

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО»**

Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

“На правах рукопису”

УДК 004.78

“До захисту допущено”

Завідувач кафедри

О.В. Коваль
(підпис) (ініціали, прізвище)

“ ” 2019р.

Магістерська дисертація

зі спеціальності 121 Інженерія програмного забезпечення
за спеціалізацією Інженерія програмного забезпечення розподілених систем
на тему “Розробка програмного агента моніторингу і управління режимами роботи мікроенергостанцій”

Виконав (-ла): студент (-ка) 6 курсу, групи ТВ-81мп

Столяр Андрій Владиславович

(прізвище, ім'я, по батькові)

(підпис)

Науковий керівник доц. к.т.н. доц. Ковальчук А.М.

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

Консультант

(назва розділу)

(вчені ступінь та звання, прізвище, ініціали)

(підпис)

Рецензент

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.

Студент _____
(підпис)

**Національний технічний університет України
“Київський політехнічний інститут ім. Ігоря Сікорського”**

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти другий, магістерський

зі спеціальності 121 Інженерія програмного забезпечення

за спеціалізацією Інженерія програмного забезпечення розподілених систем

ЗАТВЕРДЖУЮ

Завідувач кафедри

Коваль О.В.

(прізвище, ініціали)

_____ (підпис)

«_____» _____ 2019р.

**З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ**

Столяру Андрію Владиславовичу

(прізвище, ім'я, по батькові)

1. Тема дисертації “Розробка програмного агента моніторингу і управління режимами роботи мікроенергостанцій”

Науковий керівник Ковальчук Артем Михайлович к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “___” _____ 20__ року №___

2. Строк подання студентом дисертації _____

3. Об'єкт дослідження Інформаційні системи моделювання технологічних процесів мікроенергетичних станцій

4. Предмет дослідження Програмні засоби моделювання структури та сценаріїв функціонування енергетичної інфраструктури мікроенергостанцій

5. Перелік питань, які потрібно розробити _____

— проаналізувати особливості та архітектуру інформаційних систем

моделювання режимів функціонування технологічних систем;

— проаналізувати існуючі підходи до побудови систем управління мікроенерго-
станціями;

— розробити програмні моделі функціонування елементів інфраструктури
мікроенергостанцій;

— розробити засоби реалізації сценаріїв управління з урахуванням мінливості
оточуючого середовища.

6. Орієнтований перелік ілюстративного матеріалу схема програмного агента, мультиагентна модель, сценарій функціонування енергоринку, математична модель функціонування енергоринку, структура програмної системи, графічний інтерфейс.

7. Орієнтований перелік публікацій Столяр А.В., Ковальчук А.М. “Агент моніторингу та управління режимами роботи мікроенергостанцій” (збірник «Сучасні проблеми наукового забезпечення енергетики» XVII міжнародної науково-практичної конференції аспірантів, магістрів, студентів)

8. Дата видачі завдання « ____ » _____ 20 ____ р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Строки виконання етапів магістерської дисертації	Примітка
1	Отримання завдання		
2	Аналіз вимог завдання, вибір методів і засобів розв’язання поставлених задач		
3	Підготовка публікацій		
4	Підготовка доповідей на конференції		
5	Розробка програмного продукту		
6	Підготовка дисертації		
7	Захист програмного продукту		
8	Передзахист		
9	Захист		

Студент

(підпис)

Столяр А.В.
(прізвище та ініціали)

Науковий керівник

(підпис)

Ковальчук А.М.
(прізвище та ініціали)

РЕФЕРАТ

Структура та обсяг дипломної роботи. Магістерська дисертація складається зі вступу, 4 розділів, висновку, переліку посилань з 39 найменувань, 3 додатків і містить 19 рисунків, 15 таблиць. Повний обсяг магістерської дисертації складає 97 сторінок, з яких перелік посилань займає 4 сторінки, додатки — 4 сторінки.

Актуальність теми. Застосування мікроенергостанцій є економічно вигідним вирішенням проблеми глобального споживання енергетичних ресурсів. Мікроенергостанції являють собою мережі низької або середньої напруги, які можуть інтегрувати відновлювані джерела енергії, системи зберігання енергії та її споживачів. Для проектування мікроенергостанцій та забезпечення їх ефективної роботи необхідні системи моделювання та системи енергоменеджменту. Існуючі на даний момент системи енергоменеджменту переважно використовують централізовану архітектуру та орієнтовані на ручне управління. Недоліками існуючих систем моделювання є обмежені або відсутні можливості симуляції моделі та відсутність моделювання взаємодії між компонентами системи. Тому актуальною є проблема переходу до децентралізованої архітектури систем енергоменеджменту та застосування агентного підходу для їх моделювання та проектування.

Мета дослідження полягає у створенні системи моделювання мікроенергостанцій і програмного агента управління режимами їх роботи.

Для досягнення поставленої мети визначено **завдання дослідження**:

1. Проаналізувати особливості та архітектуру інформаційних систем моделювання режимів функціонування технологічних систем.
2. Проаналізувати існуючі підходи до побудови систем управління мікроенергостанціями.
3. Розробити програмні моделі функціонування елементів інфраструктури мікроенергостанцій.

4. Розробити засоби реалізації сценаріїв управління з урахуванням мінливості оточуючого середовища.

Об’єкт дослідження — інформаційні системи моделювання технологічних процесів мікроенергетичних станцій.

Предмет дослідження — програмні засоби моделювання структури та сценаріїв функціонування енергетичної інфраструктури мікроенергостанцій.

Методи дослідження. Для розв’язання поставлених задач використано методи математичного моделювання та методи об’єктно-орієнтованого аналізу.

Інноваційна новизна одержаних результатів полягає у вдосконаленні існуючих систем моделювання та енергоменеджменту за рахунок:

- застосування мультиагентного підходу для формування та управління енергетичним аукціоном і, відповідно, управління мікроенергостанцією;
- використання уніфікованої моделі генератора, що спрощує розробку та масштабування мультиагентної системи, враховуючи індивідуальні особливості генератора (вітряк, СЕС тощо) через його енергетичні характеристики;
- впровадження стратегії поведінки учасників торгів у умовах енергоринку.

Практичне значення одержаних результатів полягає в розробленій програмній системі, що направлена на підвищення експлуатаційної та економічної ефективності мікроенергостанцій.

Апробація результатів дисертації. Основні положення роботи доповідались і обговорювались на XVII-й міжнародній науково-практичній конференції аспірантів, магістрантів, студентів “Сучасні проблеми наукового забезпечення енергетики” (Київ 23-26 квітня 2019 р.).

Публікації. Тези доповіді публікувалися в збірнику “Сучасні проблеми наукового забезпечення енергетики” XVII міжнародної науково-практичної конференції аспірантів, магістрів, студентів.

Ключові слова. ПРОГРАМНИЙ АГЕНТ, МУЛЬТИАГЕНТНА СИСТЕМА, МІКРОЕНЕРГОСТАНЦІЯ, ЕНЕРГОРИНОК, НАВЧАННЯ З ПІДКРІПЛЕННЯМ.

ABSTRACT

Structure and volume of the dissertation. Master's dissertation consists of introduction, 4 sections, conclusion, list of sources of 39 items, 2 appendixes and includes 19 images, 15 tables. Whole dissertation volume is 97 pages, out of which list of sources takes 4 pages, appendixes — 4 pages.

Significance of the topic. The use of microgrids is a cost-effective solution to the problem of global energy consumption. Microgrids are low or medium voltage grids that can integrate renewable energy, energy storage systems and their customers. Modeling and energy management systems are required to design micro-power stations and ensure their efficient operation. The current energy management systems mainly use a centralized architecture and are focused on manual control. The disadvantages of existing simulation systems are the limited or no simulation capabilities of the model and the lack of interaction modeling between system components. Therefore, the problems of transition to a decentralized architecture of energy management systems and the use of an agent approach for their modeling and design are significant.

The objective is to create a microgrids simulation system and a software agent to control the modes of their operation.

In order to achieve the objective next **goals** were defined:

1. To analyze the features and architecture of modeling systems for the modes of operation of technological systems.
2. To analyze the existing approaches to the development of microgrid management systems.
3. Develop software models for the functioning of microgrid infrastructure elements.
4. Develop tools for the implementation of management scenarios taking into account environmental variability.

Research object: information systems of modeling of technological processes of microgrids.

Research subject: software tools for modeling the structure and scenarios of functioning of energy infrastructure of microgrids.

Research methods. Mathematical modeling and object-oriented analysis methods were used to solve the problems.

Innovational novelty of the results lies in the improvement of existing energy modelling and management systems by:

- application of multi-agent approach for the formation and management of the energy auction and, accordingly, the management of the micro-power station;
- the use of a unified model of the generator, which simplifies the development and scaling of the multiagent system, taking into account the individual characteristics of the generator;
- implementation of the strategic behavior of bidders in the conditions of the energy market.

Practical value lies in the developed software system aimed at improving the operational and economic efficiency of microgrids.

Approbation. The main provisions of the work were reported and discussed at the XVII International Scientific and Practical Conference of postgraduate students, masters, students “Modern problems of scientific support of energy” (Kyiv, April 23-26, 2019).

Publications. Theses of the report were published in the collection “Modern problems of scientific support of energy” of the XVII international scientific-practical conference of postgraduate students, masters, students.

Key words. *SOFTWARE AGENT, MULTI-AGENT SYSTEM, MICROGRID, REINFORCEMENT LEARNING.*

ЗМІСТ

Перелік умовних позначень і термінів	9
Вступ	10
1. Аналіз проблеми управління режимами роботи мікроенергостанцій	12
1.1 Поняття мікроенергостанції	12
1.2 Основні проблеми мікроенергостанцій	15
1.3 Системи управління мікроенергостанціями	17
1.4 Задача розробки програмного агента моніторингу та управління режимами роботи мікроенергостанцій	22
1.5 Огляд існуючого програмного забезпечення	23
1.5.1 Системи моделювання мікроенергостанцій	24
1.5.2 Системи енергоменеджменту	26
Висновки до розділу 1	28
2. Агентний підхід до управління режимами роботи мікроенергостанцій	30
2.1 Агентний підхід до моделювання складних систем	30
2.1.1 Мультиагентна система	31
2.1.2 Мультиагентне моделювання	32
2.1.3 Визначення агента	33
2.1.4 Паралелі між ABM та ООП/OOM	34
2.2 Інструменти моделювання мультиагентних систем	36
2.2.1 Платформа JADE	36
2.2.2 Платформа Repast	39
2.3 Підходи до організації управління мікроенергостанціями	42
Висновки до розділу 2	46
3. Програмна реалізація системи моделювання мікроенергостанцій	47
3.1 Програмна модель мікроенергостанції	47
3.2 Загальний опис системи	49

3.3	Опис архітектури системи.....	51
3.2.1	Огляд архітектури	52
3.2.2	Конфігурація програмних агентів	55
3.2.3	Модель даних	56
3.3	Задача про оптимальний потік енергії у мережі	56
3.4	Навчання програмних агентів	60
3.4.1	Алгоритм VRE.....	60
3.4.2	Засоби реалізації.....	63
3.5	Графічний інтерфейс користувача	64
	Висновки до розділу 3	67
4.	Стартап проект	69
4.1	Опис ідеї проекту	69
4.2	Технологічний аудит ідеї проекту.....	71
4.3	Аналіз ринкових можливостей запуску стартап проекту	73
4.4	Розроблення ринкової стратегії стартап проекту	81
4.5	Розроблення маркетингової програми стартап проекту	85
	Висновки до розділу 4	88
	Висновки	89
	Список використаних джерел	90
	Додаток А.....	94

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ

ABM — Agent Based Modelling, мультиагентне моделювання.

API — Application programming interface, програмний інтерфейс.

CEMS — Централізована система енергоменеджменту.

DAM — Day Ahead Market, алгоритм роботи енергоринку.

DEMS — Децентралізована система енергоменеджменту.

Demand Side Management — концепція менеджменту на стороні споживача.

EMS — Energy Management System, інформаційна системи енергоменеджменту.

FIPA — Foundation for Intelligent Physical Agents, організація та система стандартів для агентів та мультиагентних систем.

GUI — Graphical user interface, графічний інтерфейс користувача

IDE — Integrated development environment, вбудована система розробки.

LMP — Locational Marginal Price, локальна гранична ціна.

MAS — Multi-Agent System, мультиагентна система.

Middleware — програмне забезпечення, що слугує мостом між іншими програмами.

P2P — Peer-To-Peer, варіант архітектури системи, в основі якої стоїть мережа рівноправних вузлів.

SCADA — Supervisory Control And Data Acquisition, центральний контролер мікроенергостанції у централізованій архітектурі.

ООП — Об'єктно-орієнтоване програмування.

ООМ — Об'єктно-орієнтоване моделювання.

ВСТУП

Зростання чисельності населення, урбанізація та модернізація призвели до однієї з найбільших проблем сучасності, яка полягає у значному зростанні світового попиту на електроенергію. Економічно вигідним вирішенням проблеми глобального споживання енергетичних ресурсів є застосування мікроенергостанцій або енергетичних мікромереж.

Мікроенергостанції являють собою мережі низької або середньої напруги, які можуть інтегрувати відновлювані джерела енергії (ВДЕ), генератори, системи зберігання енергії та енергоспоживання, тобто являють собою системи енергетичної інфраструктури [1].

Однією з ключових особливостей мікроенергостанцій є їх здатність працювати як в автономному режимі, так і в режимі підключення до глобальної енергомережі [2]. Особливий економічний інтерес представляє собою саме робота у автономному режимі, оскільки більшість людей, що мають проблеми із енергозабезпеченням живуть у віддалених або острівних регіонах [3].

Для забезпечення надійної та ефективної роботи мікроенергостанцій необхідні системи енергоменеджменту (EMS), а для їх попереднього проектування — системи моделювання мікроенергостанцій.

Існуючі на даний момент системи енергоменеджменту переважно використовують централізовану архітектуру та орієнтовані на ручне управління. Основним недоліком таких систем є складність масштабування та потреба у спеціалістах-операторах [4]. Тому актуальною є проблема переходу до децентралізованої архітектури при побудові систем енергоменеджменту, та впровадження автоматизації в управління роботою мікроенергостанцій.

Недоліками існуючих систем моделювання є обмежені або відсутні можливості симуляції моделі, а також їх зосередженість виключно на економічних показниках мікроенергостанції, з чого випливає, що побудовані

моделі не враховують взаємодію між компонентами модельованої системи.

Метою роботи є створення системи моделювання мікроенергостанцій та програмного агента моніторингу та управління режимами їх роботи.

Об'єкт дослідження — інформаційні системи моделювання технологічних процесів мікроенергетичних станцій.

Предмет дослідження — програмні засоби моделювання структури та сценаріїв функціонування енергетичної інфраструктури мікроенергостанцій.

Агент моніторингу і управління режимами роботи мікроенергостанцій є одним із найважливіших компонентів системи енергоменеджменту. Агент здатен відстежувати зміни як в зовнішній енергетичній мережі, так і в мікромережі, та впливати на неї для досягнення цілей свого проектування. Паралельно взаємодіючи, координуючи або ведучи переговори з іншими агентами системи, він обробляє отриману інформацію, порівнює з певними правилами, досліджує можливі рішення і вибирає оптимальне, таким чином реалізуючи концепцію Demand Side Management.

Виконана робота включає в себе аналіз існуючих інформаційних систем енергоменеджменту та систем моделювання мікроенергостанцій, огляд алгоритмів автоматизованого управління режимами роботи мікроенергостанцій, побудову мультиагентної програмної моделі мікроенергостанції та розробку визначеного задачею програмного агента у її складі.

Розроблена система може використовуватись у будь-якому середовищі, де важливим є питання енергозбереження та оптимального використання енергоресурсів. Основна цільова аудиторія — компанії енергетичної галузі, а також кафедри енергетичних спеціальностей університетів.

1. АНАЛІЗ ПРОБЛЕМИ УПРАВЛІННЯ РЕЖИМАМИ РОБОТИ МІКРОЕНЕРГОСТАНЦІЙ

Так як розроблений продукт призначений для моделювання та управління мікроенергостанцією, перш за все потрібно розкрити це поняття. У наступних пунктах наведено огляд основних проблем, пов'язаних із мікроенергостанціями і відповідно проблеми організації їх управління, проаналізовано аналогічні програмні застосунки та сформульовано задачу розробки.

1.1 Поняття мікроенергостанції

У найзагальнішому випадку поняття мікроенергостанції, або мікроенергомережі, можна визначити як локалізовану групу джерел електроенергії та її споживачів, ключовою особливістю якої є здатність працювати як у режимі підключення до традиційної широкомасштабної (глобальної) електроенергетичної мережі, так і автономно, у так званому острівному режимі. Загальна схема мікроенергостанції зображена на рисунку 1.1.

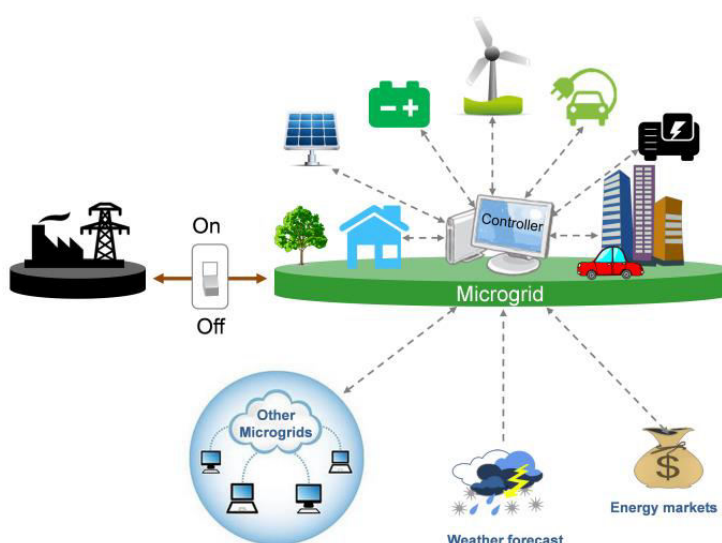


Рисунок 1.1 — Схема мікроенергостанції

У режимі, підключення до глобальної мережі, мікроенергостанція закупає електроенергію з мережі або вводить в неї локально згенеровану енергію для того, щоб урегулювати баланс потужності між споживанням та генерацією енергії та максимізувати експлуатаційні переваги мікроенергомережі як одиниці у складі глобальної мережі.

У острівному режимі мікроенергостанція працює повністю автономно і постачає своїм споживачам електроенергію, зберігаючи стабільний електричний обмін між генерацією та споживанням [5].

Сучасні мікроенергостанції за типом можна поділити на:

- наукові мікроенергостанції на базі університетів та науково-дослідних інститутів;
- громадські комунальні мікроенергостанції;
- автономні мікроенергостанції віддалених регіонів;
- військові мікроенергостанції;
- комерційні та індустріальні мікроенергостанції [6].

У центрі уваги наукових мікроенергостанцій є об'єднання та організація взаємодії між існуючими місцевими генераторами та деякою кількістю споживачів, що розташовані на відносно тісній місцевості, в якій власник має змогу легко ними керувати.

Громадські мікроенергостанції можуть обслуговувати до декількох тисяч споживачів та задовольняти потреби у електроенергії, опаленні та охолодженні. Деякі будинки у складі громадської мікроенергомережі можуть мати поновлювані енергетичні джерела, які можуть задовольнити споживання не лише їх мешканців, а також і їх сусідів в межах однієї громади. Громадська мікроенергомережа також може мати централізоване або кілька розподілених сховищ енергії. Такі мікроенергостанції можуть оперувати у режимі двох мереж: мережі змінного та мережі постійного струму, об'єднаних між собою через двонаправлений силовий електронний перетворювач [7].

Автономні мікроенергостанції ніколи не з'єднуються із глобальною

енергомережою, а натомість постійно працюють в острівному режимі через економічні проблеми або географічне положення. Як правило, такі мережі будуються в районах, віддалених від будь-якої існуючої інфраструктури передачі та розподілу. Дослідження продемонстрували, що експлуатація мікроенергомереж на віддалених територіях або островах, на яких переважають відновлювані джерела, зменшить вирівняні витрати на виробництво електроенергії протягом життя таких проектів [8].

Великі віддалені райони можуть постачатися одразу декількома незалежними мікроенергостанціями, кожна з яких має свого власника. Хоча такі мережі традиційно розроблюються для енергозабезпечення, переривчасті поновлювані джерела, а також їх несподівані та різкі зміни можуть спричинити несподіваний дефіцит електроенергії або надмірну енергогенерацію у цих мікроенергомережах. Це одразу спричинить неприйнятне відхилення напруги або частоти. Для виправлення або уникнення подібних ситуацій можливо підключити такі мережі до відповідної сусідньої мікроенергомережі для обміну енергією та поліпшення частотних відхилень та відхилення напруги. Цього можна досягти за допомогою перемикача на основі спеціальної електроніки після належної синхронізації двох електронних перетворювачів та після підтвердження стабільності нової системи. Визначення необхідності об'єднання сусідніх мереж та пошуку підходящої мережі для сполучення може бути досягнуто за допомогою застосування оптимізаційних підходів [9].

Військові мікроенергостанції є окремим підвидом автономних. Їх задача полягає у енергетичному забезпеченні військових структур, яке не залажало б від стану глобальної мережі енергозабезпечення, робота якої може бути порушена у наслідок бойових дій. Завдяки активному розвитку військових мікроенергостанцій розроблюються та впроваджуються нові способи захисту мікроенергомереж як на фізичному, так і на інформаційному рівні [10].

Основними причинами використання промислової мікроенергостанції є безпека електроживлення та її надійність. Існує багато виробничих процесів, для

яких переривання електропостачання може спричинити значні втрати доходу та призвести до тривалого повторного запуску. Промислові мікроенергостанції можуть розроблюватись для забезпечення кругової економіки, промислових процесів із близькими до нуля або нульовими відходами, і можуть включати в себе комбіновану генерацію тепла та електроенергії, живлячись як відновлюваними джерелами, так і переробкою відходів. Накопичення енергії може бути додатково використане для оптимізації роботи цих підсистем [11].

Таким чином, перевагами мікроенергостанцій або мікроенергомереж є їх здатність працювати як у автономному режимі, так і в режимі підключення до глобальної енергомережі, використання розподіленої енергогенерації, зокрема із використанням відновлюваних енергоджерел, локалізованість, яка робить їх більш стійкими до природніх катастроф, а також більш гнучка архітектура, що дозволяє використовувати новітні системи енергоменеджменту для підвищення надійності, ефективності та економічної вигоди [12].

1.2 Основні проблеми мікроенергостанцій

Окрім визначених у попередньому пункті переваг, мікроенергостанції мають ряд проблем, які повинні бути вирішені для досягнення максимальної вигоди від їх впровадження. Основними проблемами мікроенергомереж, що пов'язані із якістю енергією є незбалансованість напруги, гармоніки струму, збільшена потреба у реактивній потужності та частотні відхилення [13].

Незбалансованість напруги виникає при переході від режиму підключення до глобальної мережі до острівного режиму роботи, якщо мікроенергостанція забирає або віддає енергію у глобальну мережу у момент відключення. Зважаючи на повільний відклик та низьку інерцію деяких мікро джерел, для вирішення цієї проблеми використовуються пристрої накопичення та зберігання енергії. Коли мікроенергостанція знову з'єднується із глобальною мережею, вона синхронізується, забезпечуючи однакові величини та фази напруги по всьому

синхронізуючому пристрою. Для відключення може використовуватись високошвидкісний статичний перемикач [14].

Через розрізнені характеристики генераторів у складі мікроенергостанції виникають проблеми із стабільністю системи, порівняно із традиційним виробництвом. Зокрема: пониження кутової стабільності за рахунок пониження загальної інерції в системі, пониження стабільності напруги за рахунок низького розподілення енергії у мережі, а також низькочастотні коливання потужності [15].

Гармонічні струми мають бути мінімізовані для зменшення гармонічних втрат. Гармоніки, що виникають завдяки роботі електронних перетворювачів, гармонійна напруга і струм повинні бути обмежені прийнятним рівнем в точці підключення генератора до мережі. Для забезпечення прийнятного рівня гармонійної напруги, кожне джерело гармонічного струму може пропускати лише обмежений внесок у мережу, згідно з рекомендаціями ІЕС-61400-36 [16].

У низьковольтних мікроенергомережах, особливо у мережах, де для генерації використовуються сонячні батареї, виникає необхідність у використанні інверторів змінного струму у постійний. Із великою кількістю інверторів виникають сумніви щодо їх можливостей у підтримці системи у моменти, коли мікроенергомережа знаходиться під суттєвим навантаженням. Таке навантаження може виникнути унаслідок збоїв у системі або завдяки відхиленням напруги або частоти. Ця проблема може бути вирішена наданням додаткового функціоналу інверторам, наприклад регулювання реактивної потужності та зменшення гармонік [17].

Загалом, робота мікроенергостанцій містить дуже багато невизначеностей, що впливають на її надійність та економічну привабливість. Графік навантаження та погодні зміни — дві з них, які роблять координацію більш складною в автономних мережах, де критичним є баланс попиту та пропозиції та, а показники впливу відмови компонентів на роботу системи в цілому є більш високими. Ці невизначеності вимагають вирішення складної оптимізаційної задачі для забезпечення оптимальної роботи мікроенергостанції.

1.3 Системи управління мікроенергостанціями

Основна мета системи управління мікроенергомережею, або системи енергоменеджменту — оптимізувати роботу місцевого виробництва енергії. До функцій, якими керує система енергоменеджменту належать: прогнозування електричного навантаження, прогнозування можливостей виробництва електроенергії з відновлюваних джерел, планування генерації, скидання навантажень, розрахунки викидів, управління попитом та оцінка безпеки. Також система управління несе такі обов'язки, як забезпечення належної роботи джерел енергії у заздалегідь заданих точках роботи, задоволенні робочих меж, активних та реактивних потужностей мікроенергомережі, безперебійне підключення та відключення до зовнішньої електромережі, дозволяючи розподіленим генераторам брати участь на енергетичному ринку [18]. Таким чином, система управління мікроенергомережею або мікроенергостанцією може бути визначена як типовий представник автоматизованої системи енергетичної інфраструктури.

Автоматизована система енергетичної інфраструктури — система збору первинної інформації, аналітики та управління енергетичними ресурсами, керування роботою пристроїв для забезпечення енергоефективних режимів їх роботи. Призначення системи полягає в автоматизованому вирішенні проблем пов'язаних із використанням енергетичних ресурсів і може ефективно застосовуватись як в житлових умовах, так і на промислових підприємствах.

Якщо розглядати будь-яку сучасну мікромережу у вигляді дворівневої моделі, то система енергоменеджменту представляє собою верхній, інформаційний рівень (рисунок 1.2).

Системи енергоменеджменту мікроенергостанцій можуть бути розділені на два основні класи: централізовані (CEMS) і децентралізовані (DEMS), що, як правило, визначається централізаційними ролями, що приймаються при проектуванні системи управління мережею.

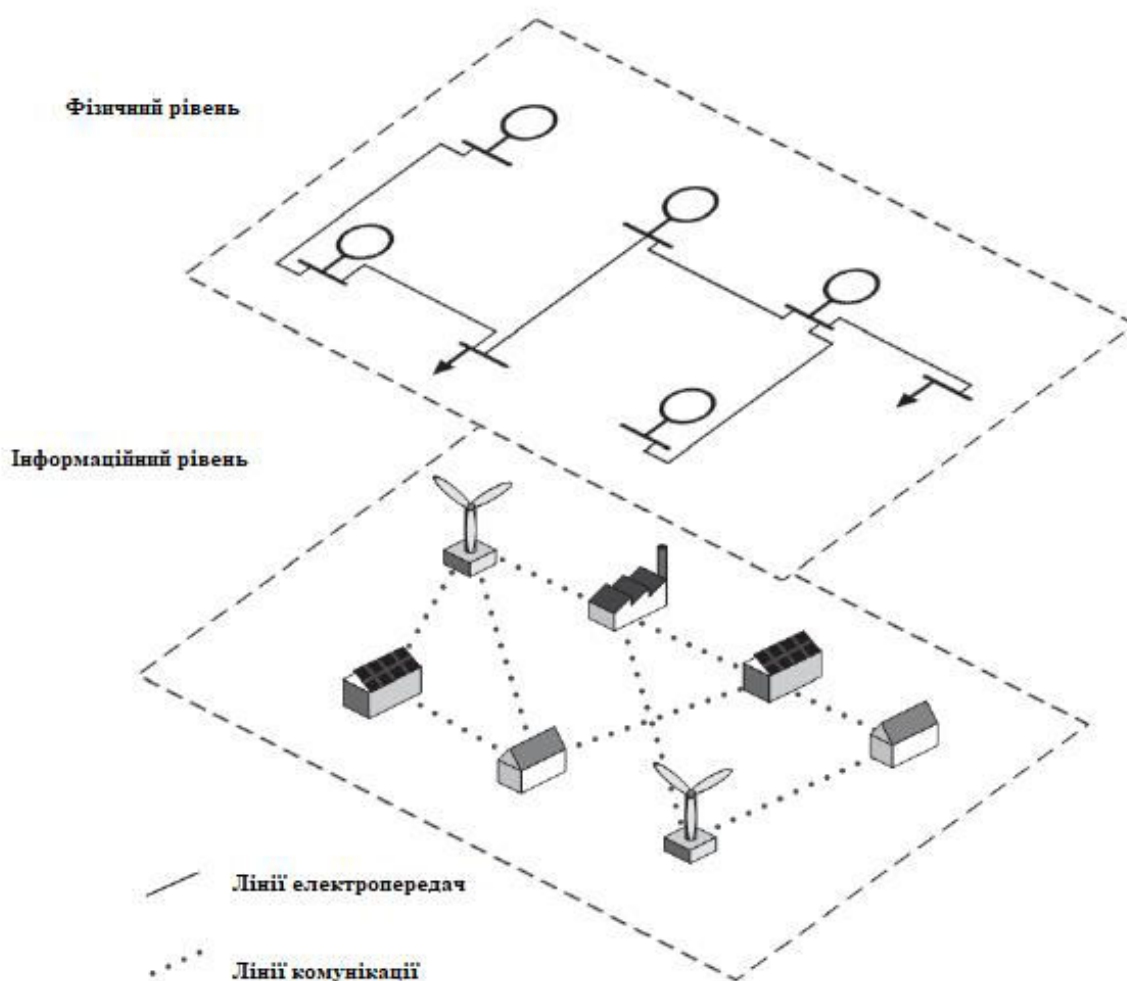


Рисунок 1.2 — Дворівнева модель мікроенергомережі

CEMS представляють собою ієрархічні системи, центральний контролер яких збирає всю необхідну інформацію із компонентів мікроенергостанції, виконує набір управляючих команд, які безпосередньо виконують централізоване функціонування, таким чином приймаючи на себе роль загального системного менеджера. Вся система, що побудована за архітектурою CEMS фактично не може функціонувати без центрального контролера.

Протягом останніх десятиліть були досліджені та застосовані різні реалізації CEMS архітектури для мікроенергомереж, маючи на меті досягнення ефективного контролю різних джерел енергії, мінімізації експлуатаційних витрат, відхідних викидів та досягнення ефективного управління енергією.

На рисунку 1.3 зображено приклад застосування CEMS архітектури, що може застосовуватись для управління розподіленими генераторами та іншими компонентами мікроенергостанції.



Рисунок 1.3 — Централізована архітектура системи енергоменеджменту

Відповідно до наведеного на рисунку приклада, центральний контролер представляє собою SCADA-компонент системи (Supervisory Control And Data Acquisition).

До переваг CEMS архітектури можна віднести спрощення спостереження за роботою системи та мінімізацію конфліктів на інформаційному рівні, оскільки всі остаточні рішення приймає центральний контролер.

Попри свої переваги, CEMS архітектури мають суттєві недоліки. Для забезпечення функціонування мікроенергостанцій, CEMS потребують обчислювальної техніки високої потужності, що здатна ефективно оброблювати величезні обсяги даних, а також надійних та широко розповсюджених ліній комунікації [19]. Це в свою чергу призводить до значних витрат на встановлення та утримання. Також централізований підхід потребує встановлення спеціального, часто пропрієтарного обладнання, що має бути встановлено на компоненти системи. Особливо небажаним це є у випадку, коли більшість розповсюджених

генераторів та елементів накопичення енергії мають невелику місткість, і відповідно встановлення окремого управляючого обладнання на кожен елемент або генератор не є економічно вигідним. Крім того, CEMS архітектура диктує низьку гнучкість, оскільки всю систему необхідно оновлювати при зміні конфігурації мережі [20].

Централізоване управління найкраще підходить для мереж, де власники генераторів і споживачів мають спільні цілі та прагнуть співпраці для досягнення своїх індивідуальних цілей та мереж, що повільно розвиваються та містять переважно генератори великої місткості.

Децентралізована або розподілена архітектура (рисунок 1.4) може бути поділена на декілька типів, від гібридних до повністю децентралізованих. DEMS архітектура є більш гнучкою і менш складною, оскільки вона базується на мережі автономних локальних контролерів.

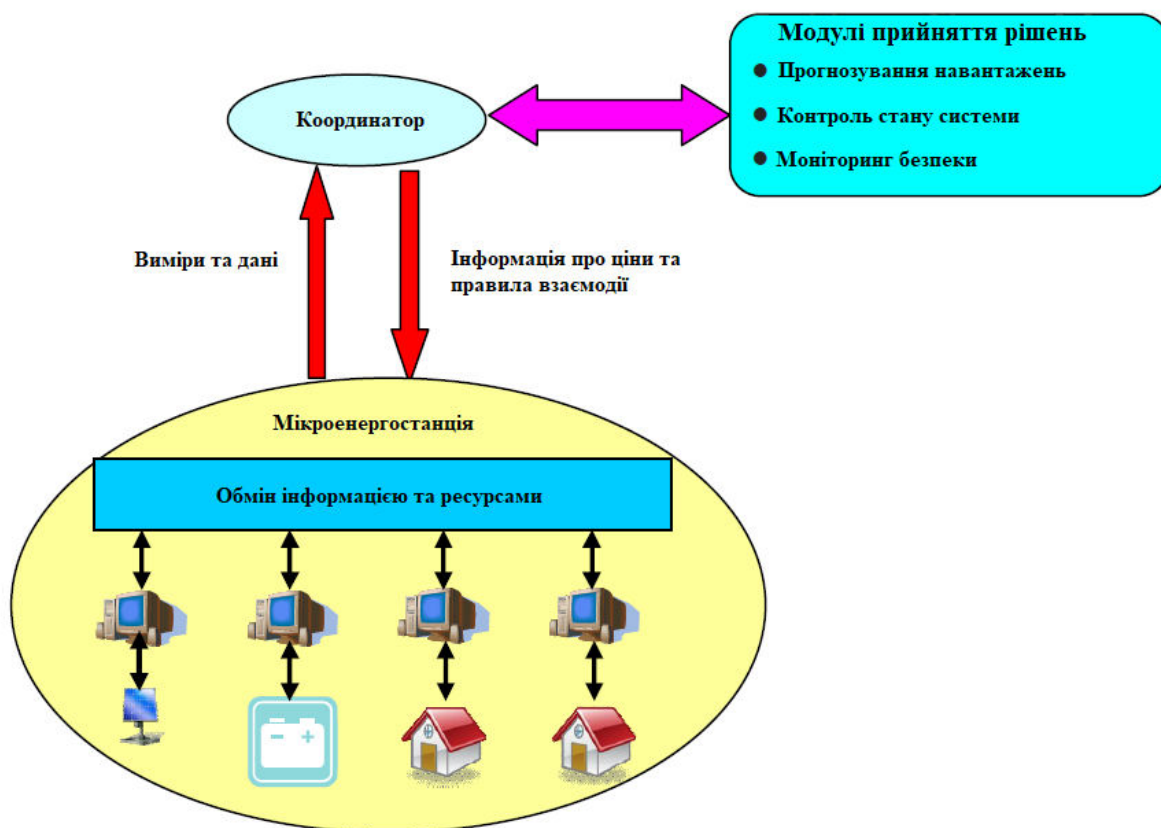


Рисунок 1.4 — Децентралізована архітектура системи енергоменеджменту

Основна відмінність між двома архітектурами полягає в обов'язках, покладених на компоненти системи. Компоненти децентралізованої системи енергоменеджменту є незалежними, вони формують інтелектуальну і динамічну систему, що забезпечує енергоефективність і відповідає змінним умовам, таким як мінливість генерації відновлюваних енергоджерел та навантаження системи з боку споживачів. Завдяки їх особливостям та розвитку технології мікроконтролерів, у DEMS можуть використовуватись передові методи управління.

У наведеному прикладі типової системи енергоменеджменту із децентралізованою архітектурою контролер, або координатор, виконує лише базові обов'язки, мінімальні для забезпечення цілісності системи. Розрахунок власних економічних показників, контроль генерації та споживання виконуються на розподілених мікроконтролерах, кожен з яких вирішує ці задачі для себе, обмінюючись інформацією із координатором та іншими контролерами.

У повністю децентралізованих системах управління мікроенергомережами кожен контролер зазвичай виконує управління одним компонентом системи. Ці контролери збирають лише локальну інформацію про пристрій, що знаходиться під їх контролем. Їм майже нічого невідомо ні про параметри системного рівня, ні про управлінницьку діяльність сусідніх контролерів доки не буде встановлено явний зв'язок із метою обміну інформацією [21]. Контролери, через свої обмежені знання, можуть приймати рішення, обґрунтовані лише на основі власної інформації та інформації, отриманої при взаємодії з сусідніми контролерами. Однак у системах, де зв'язок між різними робочими блоками вимагає певного рівня координації, повністю децентралізована архітектура не в змозі досягти стабільної роботи на основі лише місцевої інформації, що є одним із недоліків такого підходу та причиною виникнення гібридної архітектури.

Децентралізовані системи енергоменеджменту найкраще підходять для мікроенергостанцій у тих випадках, коли їх розподілені генератори мають різних власників.

1.4 Задача розробки програмного агента моніторингу та управління режимами роботи мікроенергостанцій

Метою дипломної роботи є розробка комплексного продукту, що включає в себе систему побудови мультиагентних моделей мікроенергостанцій та програмний агент управління режимами їх роботи.

Визначено та сформульовано наступні задачі:

1. Проаналізувати особливості та архітектуру інформаційних систем моделювання режимів функціонування технологічних систем.
2. Проаналізувати існуючі підходи до побудови систем управління мікроенергостанціями.
3. Розробити програмні моделі функціонування елементів інфраструктури мікроенергостанцій.
4. Розробити засоби реалізації сценаріїв управління з урахуванням мінливості оточуючого середовища.

Визначено вимоги до розроблюваної програмної системи:

— Програмна модель мікроенергомережі має представляти варіант конфігурації реальної мережі, що складається із уніфікованих агентів-генераторів та агентів-споживачів, ліній сполучення між ними, а також агента-координатора.

— Призначення агента-координатора полягає у зборі запитів на продаж або закупівлю енергії від відповідних агентів системи: агентів-генераторів, та агентів-споживачів, передачі отриманих даних у власний обчислювальний модуль, який повертає рекомендований графік режимів роботи агентів-генераторів, який задовольнив би потреби усіх агентів-споживачів, приносячи при цьому максимальний прибуток, та зворотного зв'язку із агентами з метою передачі їм обчислених рекомендацій.

— Структурно розроблювана система має бути розподілена на частини, які будуть взаємодіяти між собою. Спосіб досягнення цього має забезпечувати стабільність результуючої програми та можливість додавання нового функціоналу

до створеної архітектури для оновлення програми, не вносячи суттєвих змін в уже створені модулі.

Основний функціонал розроблюваної системи можна поділити на дві групи: функції забезпечення симуляції моделі мікроенергомережі та мультиагентної взаємодії і функції програмного агента моніторингу та управління режимами роботи.

Функції, що забезпечують симуляцію моделі:

- моделювання годинника симуляції;
- встановлення та перевірка виконання певних правил симуляції;
- підтримка динамічного додавання нових агентів у модель;
- підтримка комунікації між агентами;
- можливості конфігурування моделі;
- графічне та текстове представлення результатів симуляції.

Функції розроблюваного програмного агента у складі системи:

- спілкування із агентами системи для отримання запитів на продаж та купівлю енергоресурсів;
- валідація даних;
- математичне моделювання задачі оптимального управління режимами роботи мережі;
- вирішення змодельованої задачі;
- побудова розпорядку роботи агентів-генераторів та агентів-споживачів у складі мікроенергомережі.

1.5 Огляд існуючого програмного забезпечення

У ході дослідження предметної області було проведено пошук та аналіз уже існуючих програмних застосунків, аналогічних до розроблюваного. Оскільки задача дипломної роботи включає в себе як розробку системи моделювання мікроенергостанцій, так і програмного агента для організації їх автоматизованого

управління, в рамках аналізу розглянуто два види програмного забезпечення: системи моделювання мікроенергостанцій та системи їх управління (системи енергоменеджменту).

1.5.1 Системи моделювання мікроенергостанцій

Зважаючи на активний розвиток технологій, пов'язаних із мікромережами, та зростання попиту на їх впровадження, протягом останніх років розробка програмного забезпечення для моделювання таких систем стала економічно вигідною, а отже на ринку з'явилась певна кількість гравців, що пропонують продукти промислової якості. Найвідомішими системами моделювання мікроенергосистем на ринку є комерційні проекти Distributed Energy Resources Customer Adoption Model (DER-CAM), HOMER Energy, LEAP, EAM, PV-Design Pro [22]. У якості типових представників даного класу програмного забезпечення для аналізу обрано DER-CAM та HOMER.

DER-CAM розроблено Національною лабораторією Берклі Лоуренс у Берклі, США у 2000 році. DER-CAM — це інструмент для підтримки прийняття рішень проектування децентралізованих енергетичних систем. Основна мета DER-CAM — проводити техніко-економічну оцінку визначених технологій місцевого вироблення енергії, систем опалення або мікроенергостанцій, та оптимізувати вибір розподілених генераторів та режимів їх роботи за допомогою методів лінійного програмування. Основна мета DER-CAM — мінімізація експлуатаційних витрат на локалізоване вироблення енергії.

Вхідні дані цього інструмента включають профілі навантаження, тарифи споживачів на електроенергію, ціну на природний газ, початковий інвестиційний капітал, експлуатаційні витрати, основні фізичні характеристики використовуваних генераторів та технологій рекуперації та охолодження, а також параметри обмеження викидів вуглецю та чутливості системи. DER-CAM визначає оптимальні технології для організації розподіленої генерації та системи опалення, із їх відповідним рівнем потужності, оптимальною стратегією

розподілу та загальною вартістю виробництва електричної та теплової енергії [23]. Вихідна інформація представлена у форматі таблиць та графіків (рисунок 1.5).

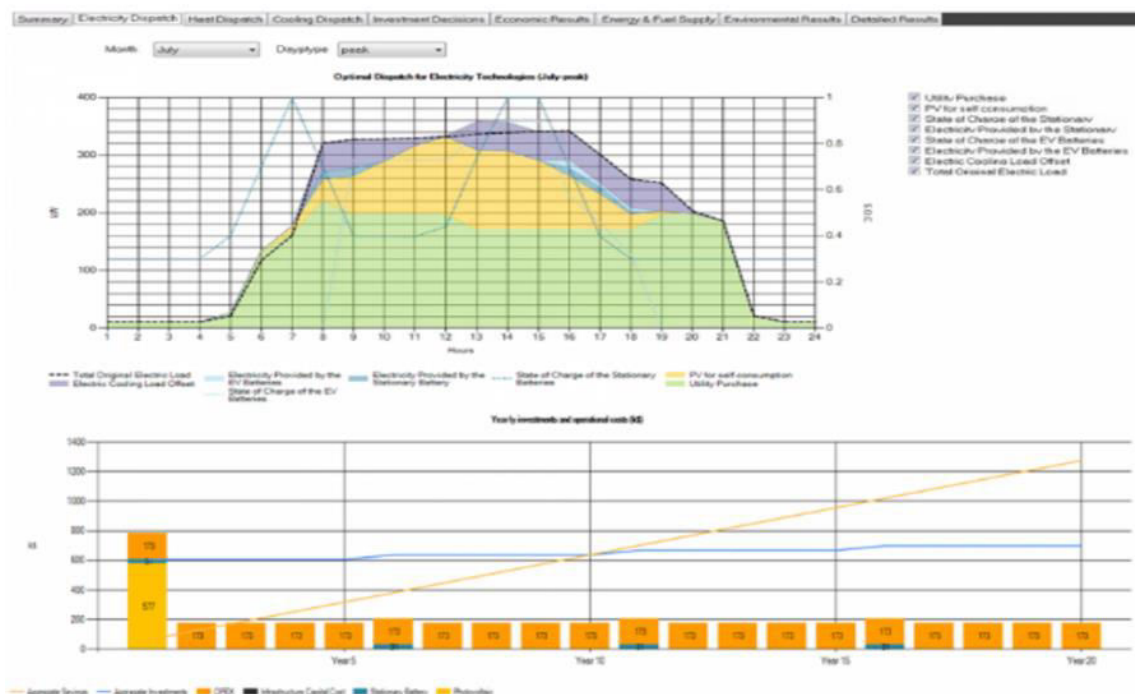


Рисунок 1.5 — Результати обчислень DER-CAM

Пакет HOMER розроблено Національною лабораторією з відновлюваної енергії (NREL). HOMER розроблено із метою оцінки мікроенергостанцій, що працюють автономно та в режимі підключення до глобальної мережі, з економічної та інженерної точки зору. Однією із сильних сторін пакету є можливість симуляції систем, однак основним призначенням є саме аналіз невизначеностей та чутливих місць модельованої системи, а також економічна оптимізація. Оптимізація проводиться на змінних, над якими має контроль дизайнер системи, а аналіз чутливості проводиться на змінних, що навпаки є невизначеними або мінливими, контроль над якими не підвласний дизайнеру, наприклад швидкість вітру або ціна на паливо [24].

Вхідні дані HOMER включають профілі навантаження споживача на електричну та теплову енергію, будь-які ресурси, використовувані системою для

отримання електричної та теплової енергії, компоненти енергетичної системи (генерація, розподіл, зберігання тощо), криві електричного та теплового навантаження із роздільною здатністю до однієї хвилини, криві технічної ефективності, експлуатаційних витрат, обмежень на викиди та параметри мінливості. Пакет HOMER видає результати оцінки та аналізу у форматі графіків та детальних звітів про дані.

Ознайомившись із даним типом програмного забезпечення, можна зробити висновок, що при розробці продукту, описаного задачею дипломної роботи, варто орієнтуватись саме на нього, враховуючи недоліки існуючих реалізацій. Так, найбільш близьким за функціоналом є пакет HOMER, а його недоліком, порівняно із розроблюваною у рамках дипломної роботи системою, є обмежені можливості симуляції системи, а також відстороненість моделювання від реальної архітектури мікроенергомережі, побудованої із застосуванням мікроконтролерів. Цих недоліків можна уникнути, використовуючи мультиагентний підхід для моделювання та симуляції.

1.5.2 Системи енергоменеджменту

Системи енергоменеджменту представляють собою продукти промислового рівня, призначенням яких є моніторинг, контроль та оптимізація роботи підконтрольних енергетичних систем. Системи енергоменеджменту часто тісно пов'язані із певним апаратним обладнанням і орієнтовані на ринок, що розвивається дещо повільніше за ринок інформаційних технологій. Відповідно, пошук аналогів серед даної категорії програмного забезпечення є досить складним. У якості типового представника обрано продукт Zenon EMS.

Zenon EMS розроблено із метою контролю та диспетчеризації енергетичних систем для поліпшення їх ефективності. Основними функціями продукту є прогнозування навантажень та генерації на короткий період, автоматичний контроль та усунення пікових навантажень, організація стабільної роботи системи у наперед заданому оператором режимі.

Модуль Управління енергією (EMS) представляє профільну інформацію у зручному для користувача вигляді (рисунок 1.6), надає можливість проведення відповідного аналізу і прогнозування в онлайн режимі, передає консолідовані дані на верхній рівень системи автоматизації.



Рисунок 1.6 — Графічний інтерфейс Zenon EMS

Цей інструмент розраховано на виявлення слабких місць системи та зон потенційної оптимізації. Відповідна конфігурація EMS забезпечує оптимальне протікання процесу при пікових навантаженнях в мережі. Для електричного сектора аналіз зазвичай виконуються за 15 хвилин, а для газових мереж – цей діапазон може бути розширено до 60 хвилин. На основі встановлених критичних значень Zenon EMS, дозволяє повністю автоматизувати процес відключення кінцевих користувачів або активізації додаткових джерел енергії. Результати оптимізаційного моделювання, здійснені на базі цього інструменту можуть бути негайно впроваджені в системі автоматизації [25].

Основним недоліком Zenon та систем енергоменеджменту загалом є відмінність у підході до управління підконтрольними системами, оскільки за своєю сутністю вони орієнтовані на системи із централізованим контролем. Із цього також випливає орієнтованість на кінцевого користувача — оператора системи, якому продукти даного типу надають асистування та автоматизацію певних управлінницьких процесів. Такий підхід суперечить задачі розробки, що полягає у моделюванні повністю автоматизованої системи. До інших недоліків можна віднести також високу ціну та єдиний варіант покупки продукту через підписання договорів із постачальниками програмного забезпечення даного типу, що зумовлено промисловим ринком, а також прив'язку до конкретного апаратного обладнання, часто дорогого та невідповідного для застосування у мікроенергостанціях.

Відповідно, можна зробити висновок, що при вирішенні задачі дипломної роботи слід орієнтуватись в першу чергу на програмне забезпечення першого розглянутого типу, тобто на системи моделювання мікроенергостанцій.

Висновки до розділу 1

В результаті аналізу проблеми управління мікроенергостанціями, виділено напрямки досліджень для вирішення цієї проблеми, обґрунтовані об'єкт та предмет дослідження.

Об'єкт дослідження: інформаційні системи моделювання технологічних процесів мікроенергетичних станцій.

Предмет дослідження — програмні засоби моделювання структури та сценаріїв функціонування енергетичної інфраструктури мікроенергостанцій.

Визначено та сформульовано наступні задачі:

1. Проаналізувати особливості та архітектуру інформаційних систем моделювання режимів функціонування технологічних систем.
2. Проаналізувати існуючі підходи до побудови систем управління

мікроенергостанціями.

3. Розробити програмні моделі функціонування елементів інфраструктури мікроенергостанцій.

4. Розробити засоби реалізації сценаріїв управління з урахуванням мінливості оточуючого середовища.

Проведено огляд та аналіз аналогічного програмного забезпечення на ринку, визначено їх недоліки, яких слід уникнути при розробці програмного продукту:

— існуючі на даний момент системи енергоменеджменту переважно використовують централізовану архітектуру та орієнтовані на ручне управління, їх основними недоліками є потреба у потужних обчислювальних ресурсах та системах комунікації, складність масштабування та потреба у спеціалістах-операторах;

— недоліками існуючих систем моделювання є обмежені або відсутні можливості симуляції моделі, а також їх зосередженість виключно на економічних показниках мікроенергостанції, з чого випливає, що побудовані моделі не враховують взаємодію між компонентами модельованої системи.

2. АГЕНТНИЙ ПІДХІД ДО УПРАВЛІННЯ РЕЖИМАМИ РОБОТИ МІКРОЕНЕРГОСТАНЦІЙ

В першу чергу необхідно визначити поняття агентного підходу до моделювання систем та програмні інструменти для його реалізації. Наступним кроком є огляд наукової літератури із метою визначення та класифікації підходів до організації моніторингу та управління режимами роботи мікроенергостанцій, що базуються саме на мультиагентних моделях проектування.

2.1 Агентний підхід до моделювання складних систем

У фізичному розумінні модель представляє собою спрощене зображення реального об'єкта із відокремленням його основних ознак та відкиданням другорядних. Програмні моделі можна розглядати як метафору фізичних моделей. Зазвичай вони представляють собою концептуальні моделі — моделі, що представлені не конкретним, фізичним поданням, а у вигляді абстрактного опису реальності, що потребує представлення.

Завдяки використанню комп'ютерів, моделі можуть представляти реальний об'єкт у віртуальному, обчислювальному просторі, де вони реалізуються за допомогою спеціально розроблених засобів, таких як формалізми або мови програмування. Це, відповідно, дозволяє виконувати розрахунки, які було б надто важко виконати вручну.

Симуляційна модель — це особливий тип моделей, що здатні розвиватися із часом. У цьому контексті, симуляція полягає у змушенні моделі змінювати свій стан із плином часу. Не кожна модель підготовлена до подібних задач.

Метою дипломної роботи є побудова саме симуляційної моделі, що стане вдосконаленням існуючих на ринку подібних систем. Загалом, як провідний засіб

створення симуляційних моделей складних систем себе зарекомендували мультиагентні системи (MAS) та, відповідно, мультиагентне моделювання (ABM). Мультиагентні системи дозволяють відображати більшість динамічних змін, що відбуваються у реальній системі та досліджувати їх ефект у віртуальному середовищі.

2.1.1 Мультиагентна система

Мультиагентні системи є найбільш загальною концепцією мультиагентного підходу до моделювання систем. Зазвичай це системи, що складаються із декількох агентів з різним ступенем інтелекту, що мають обмежені знання про середовище та можуть взаємодіяти між собою певним чином (рисунок 2.1).

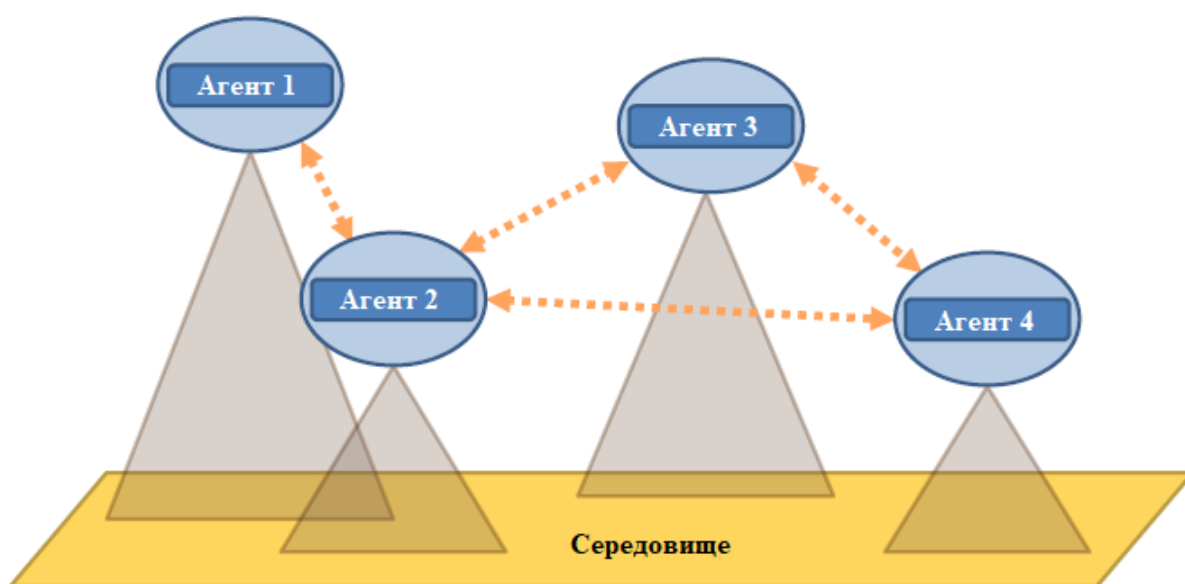


Рисунок 2.1 — Схема типової MAS

Мультиагентною можна вважати систему, що складається із наступних елементів:

- середовища, що має певний простір та ємність;
- набору об'єктів, що розташовані у середовищі, тобто у кожен момент часу може бути встановлена відповідність між об'єктом та його розміщенням у

середовищу;

- групи агентів, які є підвидом об'єктів та представляють собою активні сутності у складі системи;

- набору зав'язків, що поєднують об'єкти (та відповідно агенти) між собою;

- набору доступних операцій, що дозволяють агентам вести спостереження за об'єктами, створювати їх, використовувати та маніпулювати ними;

- операцій, що відображають застосування цих операцій та реакцію середовища на нього, тобто представляють собою закони всесвіту системи [26].

Ці характеристики показують, що MAS є безпосередньо пов'язаними зі складними системами, за рахунок їх спільних рис.

2.1.2 Мультиагентне моделювання

Мультиагентні моделі є основою розробки програмного забезпечення для моделювання складних систем. Вони є незалежними від конкретної предметної області їх застосування.

Відповідно до літератури, мультиагентне моделювання (Agent-Based Modelling) — це спосіб моделювання динаміки складних та адаптивних систем. Такі системи зазвичай здатні до самоорганізації та утворення чергового порядку. Мультиагентні моделі часто включають моделі поведінки та використовуються для спостереження за колективними ефектами поведінки агентів та міжагентної взаємодії. Розвиток інструментів мультиагентного моделювання, наявність мікроданих та успіхи в обчислювальних технологіях зробили можливим зростання кількості програмних додатків, що застосовують мультиагентний підхід [27].

Фактично, АВМ зосереджується на моделюванні систем на локальному рівні через визначення їх елементарних одиниць (агентів) та взаємодії між ними. Так, АВМ можна вважати формою моделювання мультиагентних систем.

Елементарні одиниці системи моделюються якомога більш простими, а складність системи впливає із взаємодій між ними. Доступні елементарним одиницям (агентам) дії можна поділити на три групи: спостереження за середовищем, прийняття рішень та відповідь на зміни у середовищі.

Оскільки агенти визначаються за допомогою локальної інформації, важлива різниця між мультиагентними та іншими видами моделей полягає у їх неоднорідності. Кожен агент моделюється однаково, але вхідні параметри та його локальне середовище можуть бути різними. Тому рішення, прийняті агентами можуть відрізнятися, навіть якщо процес міркування та їх цілі проектування є однаковими. Це важлива відмінність від традиційних моделей, де кілька об'єктів одного типу зазвичай розділяють однакові вхідні дані.

ABM має на меті визначення агентів найбільш простим способом для забезпечення можливості легко проводити їх масове копіювання. Так, структура системи може динамічно змінюватись через додавання або видалення окремих агентів або груп агентів, навіть під час симуляції. Це дозволяє представлення довгострокових сценаріїв, коли структура системи знаходиться в процесі неперервної зміни.

2.1.3 Визначення агента

Агенти є найважливішою частиною MAS та ABM, особливо у порівнянні із іншими, не індивідуально-центричними підходами.

Агенти — це автономні обчислювальні сутності, що можуть розглядатись як такі, що сприймають своє оточення за допомогою датчиків та діють на нього за допомогою ефекторів [28]. На рисунку 2.2 показана схема агента відповідно до даного визначення.

Агент — це реальна або віртуальна сутність, що оперує у середовищі, поведінка якої є автономною. Агент здатний сприймати, діяти та вступати у взаємодію з іншими агентами [29].

Для того, щоб назвати деяку віртуальну або реальну сутність агентом, вона

повинна відповідати хоча б базовим властивостям: самостійності у своїх діях (рішення про дії приймаються агентом а не передаються із зовні) та здатності взаємодіяти із оточення або іншими агентами.



Рисунок 2.2 — Схема програмного агента

Більше того, у розпорядженні агента знаходиться лише локальна інформація, що певним чином поєднана із середовищем. Інші властивості, такі як проактивність або інтелект є типовими для агентів, але не обов'язковими. Важливим є те, що агент був спроектований таким чином, щоб забезпечити ці розширені функції у разі потреби. Важливо розширити визначення агента у такий спосіб, оскільки це дозволяє представлення комплексних об'єктів реального світу у вигляді агентів, навіть у випадку виконання досить простих задач.

3.1.4 Паралелі між АВМ та ООП/ООМ

Об'єктно-орієнтоване програмування (ООП) — парадигма програмування, згідно з якою більше уваги приділяється предметам (об'єктам) а не на діям. Перед введенням ООП програма розглядалася як логічна процедура, що обробляє дані для створення результатів. ООП відхилилось від цієї точки зору, надаючи більшого значення правильному визначенню даних, а не програмуванню логіки.

Об'єкти визначаються через їх відносини. Абстракція забезпечує класифікацію та організацію об'єктів і даних. Відповідно до цього, в ООП зазнали широкого розповсюдження такі поняття, як узагальнення (успадкування) та обробка різних типів даних за допомогою одного інтерфейсу (поліморфізм).

Системи, побудовані у відповідності до АВМ, поділяють багато властивостей із об'єктами, визначеними об'єктно-орієнтованим програмуванням та моделюванням (ООП/ООМ). Інкапсуляція — це один із основних принципів ООП, що описує обмеження доступу до деяких компонентів об'єкта, та застосовується для мультиагентних систем аналогічним чином. Розподіл відповідальності описує процес поділу структури програмного забезпечення на різні модулі, що якомога менше перекриваються у своїй функціональності. Тобто, кожен модуль має свої чітко визначені функції, відмінні від функцій інших модулів. У мультиагентних системах агент також має свої специфічні цілі, а для різних задач зазвичай використовуються різні типи та екземпляри агентів.

Важливим моментом є те, що агенти грають активну роль у вирішенні чітко визначених задач, тобто можна сказати, що агенти представляють собою більш інтелектуальні об'єкти. Виходячи з цього, а також із певним рівнем автономії агентів, відрізняють їх від більш статичних об'єктів, описаних парадигмою ООП. Класи ООП/ООМ зазвичай також мають свої чітко визначені проектуванням задачі, але не виконують проактивну роль у їх вирішенні. Так, об'єкти надають можливості для вирішення задач, у той час як агенти їх вирішують.

Взаємодія між агентами зазвичай є складнішою, ніж взаємодія між об'єктами. Чудовим прикладом цієї відмінності є обмін повідомленнями. В ООП взаємодія полягає у виклику методів у об'єктів, що виконують певні операції та повертають певний наперед запрограмований результат. Обмін повідомленнями, що застосовуються для комунікації між агентами є більш природнім методом спілкування. Це схоже на виклик метода у об'єкта, але відмінність полягає у тому, що отримувач повідомлення є відповідальним за відповідні дії і може сам вирішувати як йому реагувати на повідомлення. Обмін повідомленнями також

надає можливість впровадження таких механізмів комунікації, як трансляція (broadcasting), що полягає у відправленні повідомлення усій системі, і лише агенти, які зацікавлені у цьому повідомленні відредагують на нього, залежно від їх локальних правил або стану.

З точки зору реалізації обмін повідомленнями також є більш складним методом комунікації. Але в той же час він пропонує більшу гнучкість та є більш адаптованим для моделювання реальних систем.

2.2 Інструменти моделювання мультиагентних систем

Зважаючи на складність програмної реалізації певних елементів мультиагентних систем, таких як комунікація між агентами, забезпечення їх цілісності та автономності, а також створення належного віртуального середовища, протягом останніх десятиліть було розроблено ряд інструментів для вирішення цих проблем. До найбільш популярних та багатофункціональних можна віднести такі продукти як NetLogo, JADE та Repast [30].

Попередньо ознайомившись із базовими властивостями цих інструментів, було визначено, що для побудови моделі мікроенергостанції найбільш доцільно використовувати інструменти JADE, або Repast. Таким чином, постало питання порівняння цих продуктів.

2.2.1 Платформа JADE

JADE (Java Agent Development Platform) — це програмна платформа, яка забезпечує основні функціональні можливості middleware, що не залежить від конкретної предметної області, та яке спрощує реалізацію програмного забезпечення, що базується на мультиагентному моделюванні.

Суттєва перевага JADE полягає в тому, що вона реалізує можливості мультиагентного моделювання та абстракцію агента для широко застосованої об'єктно-орієнтованої мови Java.

Ключовими рішеннями при проектуванні JADE було обрано наступні: агент JADE є автономним та проактивним, агенти здатні відмовити іншим агентам та є слабо пов'язаними, а система в цілому відповідає вимогам, що ставляться до P2P-архітектури.

Згідно з цими дизайнерськими рішеннями, JADE має наступні особливості:

- JADE є повністю розподіленою системою, складовими якої є агенти, кожен з яких працює у окремому потоці, потенційно на різних віддалених машинах, які здатні до прозорого спілкування один з одним; тобто платформа забезпечує API, що є абстрагованим від реальної комунікаційної інфраструктури;

- платформа повною мірою відповідає специфікаціям FIPA [31]; це стало можливим завдяки суттєвому активній співпраці розробників JADE та специфікації FIPA;

- ефективна передача асинхронних повідомлень через прозорий програмний інтерфейс; платформа обирає найкращий із доступних засобів комунікації, та за можливості уникає маршалінгу або демаршування об'єктів Java: при перетині границь платформ повідомлення автоматично перекладаються із свого об'єктного відображення у повідомлення, виконані із відповідністю до описаних у FIPA синтаксису, кодування та транспортного протоколу;

- у JADE реалізовані як білі, так і жовті сторінки, тобто агенти у складі системи можуть за потреби виконувати пошук у середовищі;

- просте, але ефективне управління життєвим циклом агента: при створенні агентів вони автоматично отримують унікальний глобальний ідентифікатор (GUID) та транспортну адресу, що використовується для реєстрації агента у службі білих сторінок; розробнику також надаються прості програмні інтерфейси та графічні інструменти для локального та віддаленого управління створенням, запуском, вимкненням та видаленням агентів;

- підтримка мобільності для агентів: і код агента, і його стан можуть мігрувати між процесами та машинами; міграція агента є прозорою для агентів у стані комунікації, що можуть продовжувати спілкування навіть під часу процесу

міграції;

- механізм підписки для агентів та зовнішніх програм, що можуть підписатись на сповіщення про події платформи;

- підтримка онтологій та специфічних для предметної області мов: платформа виконує автоматизовану перевірку, кодування та декодування, а розробники мають змогу розширювати існуючі та створювати нові мови та онтології;

- бібліотека протоколів управління, що моделюють типові моделі спілкування, орієнтовані на досягнення однієї або декількох цілей;

- інтеграція із різними веб-технологіями, включаючи сервлети, аплети та веб-сервіси;

- підтримка міжпроцесної взаємодії: наявність інтерфейсу для управління платформою та її розподіленими компонентами із сторонньої програми.

Архітектурно платформа JADE складається із контейнерів, що вміщують в собі агенти та можуть бути розподілені по мережі (рисунок 2.3).

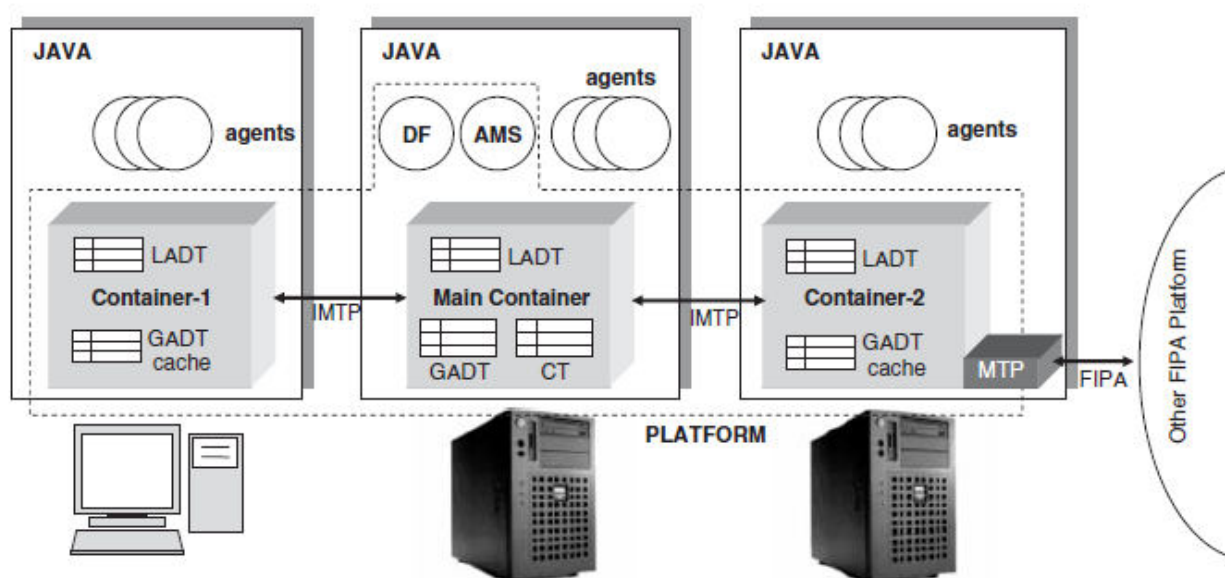


Рисунок 2.3 — Зв'язки між основними архітектурними елементами JADE

Контейнери представляють собою процеси віртуальної машини Java, що

надають агентам усі JADE та Java сервіси, що необхідні для їх створення та оперування. Головний контейнер (main container) є спеціальним і представляє собою точку запуску платформи. Головний контейнер запускається першим, а всі інші контейнери платформи реєструються у ньому. Під час запуску головного контейнера, автоматично запускаються два спеціальні агенти, ролі яких описані у стандарті FIPA для менеджменту агентів: Agent Management System (AMS) та Directory Facilitator (DF).

AMS — це агент, що контролює всю платформу. Агент є контактною точкою для всіх агентів, яким необхідний доступ до сервісу білих сторінок або менеджменту життєвого циклу. Кожен агент JADE має бути зареєстрований із AMS для того щоб отримати валідний ідентифікатор.

DF агент в свою чергу реалізує сервіс жовтих сторінок для системи. До нього звертаються усі агенти, що бажають зареєструвати свої послуги або виконати пошук доступних послуг у системі. DF також відповідає за сповіщення агентів про появу нових або про зміну існуючих послуг [32].

Перевагами платформи JADE є прозора та гнучка архітектура, підтримка розподілу системи по мережі, включаючи застосування бездротових технологій та відповідність стандартам FIPA, а основним недоліком, що призвів до відмови від даної платформи та переходу на Repast, є відсутність інфраструктури для симуляції побудованих мультиагентних моделей. Це означає, що для побудови та симуляції відносно великої мікроенергомережі, необхідно було б вручну створювати кожен агент, а також створювати власні засоби для контролю симуляції. Таким чином, JADE в першу чергу розрахована для побудови реальних мультиагентних систем, а не для моделювання або симуляції.

2.2.2 Платформа Repast

Repast (REcursive Porous Agent Simulation Toolkit) — доступний інструментарій для побудови та симуляції мультиагентних моделей. Розроблений Девідом Саллахом та іншими дослідниками Чиказького університету та

Національної лабораторії Аргонна, Repast зараз управляється некомерційною волонтерською організацією ROAD (Repast Organisation for Architecture and Development).

Repast дозволяє проводити систематичні дослідження складних систем та їх поведінки шляхом проведення контрольованих та повторюваних обчислювальних експериментів. Платформа Repast надає основний набір класів для побудови та симуляції мультиагентних систем та зручного збору результатів симуляції.

Repast може працювати практично на всіх сучасних обчислювальних платформах (Microsoft Windows, Apple Mac OS X та Linux) та має реалізації на багатьох сучасних мовах програмування, але основною реалізацією є RepastJ для мови Java.

Типова модель Repast містить набір агентів. Ці агенти можуть бути однорідними або неоднорідними, або розташованими у ієрархії. Незалежно від їх складу, кожен агент має свою поведінку, вплив якої на систему є метою досліджень. Модель Repast містить годинник симуляції (механізм планування), що контролює роботу агентів у відповідності до заданого графіку роботи. Годинник симуляції використовується не лише агентами, а і самою моделлю для збереження даних та інших утилітарних функцій.

Особливістю Repast є можливість легкої модифікації моделі та автоматизованого створення великої кількості агентів, що є необхідністю для симуляції складних систем. Repast ставить дуже мало обмежень для агентів. Так відповідно відсутні будь-які обмеження щодо гніздування агентів. Агенти все ще можуть виконуватися автономно, незалежно від того включені вони у склад комплексного агента чи ні. Агенти Repast також можуть бути змінені під час виконання моделі. Можливо також вносити зміни у модель та мережу.

Repast надає абстракції для більшості ключових елементів мультиагентного моделювання у вигляді прозорого програмного інтерфейсу на мові Java. Repast також надає користувачам графічні інструменти для створення моделей, їх запуску та керуванням параметрів.

Repast підтримує обмін даними між агентами. Механізм обміну даними полягає у обміні повідомленнями, що представлений відповідним програмним інтерфейсом. Наразі триває робота над підтримкою групового спілкування між агентами [33].

Загалом Repast надає розробникам той самий функціонал для побудови мультиагентних систем, що і JADE, а також відсутній у JADE механізм симуляції. Суттєвою відмінністю є орієнтованість Repast в першу чергу на моделювання із чого випливає відсутність відповідності стандартам FIPA. Також, на відміну від JADE, модель Repast не може виконуватись розподілено на декількох машинах, а також не підтримує онтології. Відмінності між платформами підсумовано в таблиці 2.1.

Таблиця 2.1 — Порівняння платформ JADE та Repast

	JADE	Repast
Розподілена	Так	Ні
Засоби симуляції	Ні	Так
Можливості розширення	Обмежені	Високі
Особливості роботи агентів	Програмування поведінки, багатопоточність, асинхронність, базуються на подіях	Відповідно із механізмом симуляції, багатопоточність
Міжагентна взаємодія	FIPA ACL	Обмін повідомленнями та ресурсами
Онтології	Так	Ні

В результаті порівняння двох платформ було зроблено висновок, що недоліки Repast не є суттєвими, на відміну від переваг, основними з яких є висока

розширюваність мультиагентної моделі та механізм симуляції. Завдяки цьому розроблений із застосуванням платформи програмний продукт, відповідно до поставленої задачі, дозволить легко конфігурувати різні моделі мікроенергостанцій та запускати симуляцію їх роботи.

2.3 Підходи до організації управління мікроенергостанціями

Наступним кроком є визначення способу досягнення автоматизованого управління режимами роботи модельованої мікроенергостанції. Алгоритм організації управління в першу чергу залежить від архітектури системи енергоменеджменту. Так, у централізованих системах енергоменеджменту (CEMS) зазвичай застосовуються алгоритми, що базуються на обробці певної кількості даних центральним контролером системи та відповідно прийняттям рішень про ввімкнення, вимкнення або зміну конфігурації тих чи інших компонентів системи. Такі алгоритми зазвичай застосовуються у системах енергоменеджменту для розумних будівель. У якості прикладу алгоритму для CEMS можна розглянути Energy Management Algorithm for Smart Home with Renewable Energy Sources, запропонований у Ілдізському Технічному Університеті [34]. Схема алгоритму показана на рисунку 2.4.

Відповідно до схеми, алгоритм полягає у зборі даних про стан системи з датчиків із певним інтервалом, визначенні джерела живлення та порівняння отриманих даних із набором правил. У результаті порівняння головний контролер системи посилає розміщеним на обладнанні контролерам відповідні сигнали.

Попри його простоту, випробування алгоритму показали підвищення ефективності системи на 25%. Недоліком запропонованого алгоритму як типового представника сімейства алгоритмів управління у CEMS є його низька здатність до розширення. Зі збільшенням кількості вхідних даних постане необхідність введення в систему нових правил, а також суттєво збільшиться навантаження на центральний контролер.

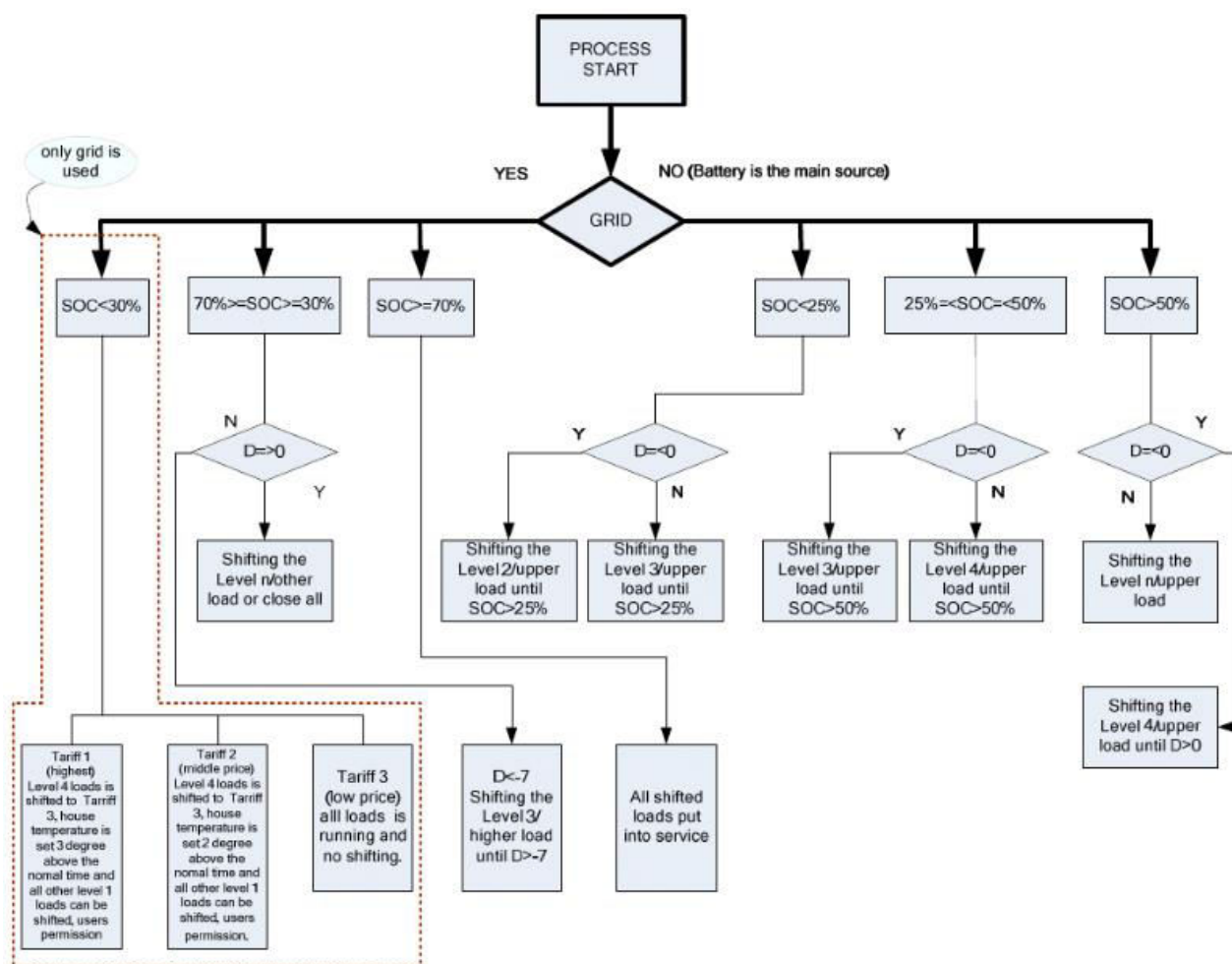


Рисунок 2.4 — Схема типового алгоритму для CEMS

Важливо зазначити про недоцільність застосування подібних алгоритмів у DEMS, тому для DEMS запропоновані алгоритми, що базуються в першу чергу на можливостях взаємодії між компонентами системи (або агентами). Відповідно, для забезпечення автоматизованого управління режимами роботи енергетичної системи у децентралізованих системах енергоменеджменту застосовуються створення та підтримка функціонування різних варіацій енергоринків.

Енергоринки як засіб організації управління в енергетичних системах відповідають традиційній моделі ринку, рушійною силою якої є попит та пропозиція. Таким чином, агенти-споживачі мають певний попит на пропозицію агентів-генераторів. Фундаментальним припущенням для моделі енергоринку є мінімізація вартості оперування системи в цілому.

Енергоринок представляє собою платформу, де учасники можуть прозоро обмінюватися електроенергією відповідно до ціни, яку вони будуть платити або отримувати, та відповідно до можливостей електричної мережі.

У літературі запропоновано способи організації енергоринків, що засновані на різних видах аукціонів. Виходячи із успішного застосування у енергоринку глобальної енергетичної мережі, доцільно розглянути аукціон закритого типу першої ціни. У даному аукціоні ціни відомі лише аукціонеру (оператору ринку), а пермагає той, хто пропонує найвищу ціну.

Як типовий приклад алгоритму управління децентралізованою енергетичною системою, в основі якого покладено аукціон закритого типу, можна розглянути Market Clearing Algorithm for the Day-Ahead Energy Market [35].

Учасниками ринку є продавці — компанії, що генерують енергію, та покупці — споживачі або компанії-дистриб'ютори енергії. Відповідно до даної моделі ринку, задачею ринкового оператора є задоволення усіх покупців та продавців на кожній ітерації.

Кожна пропозиція складається з максимальної ціни покупки та кількості енергії. За будь-яку годину та для будь-якого учасника ринку, ціни повинні збільшуватися (зменшуватися) від блоку до блоку у випадок продажу (купівлі). Пропозиції продажу можуть включати додаткові умови, такі як: обмеження на мінімальну та максимальну кількість покупки, а також мінімальний їх дохід. Пропозиції покупки додаткових умов не включають. Після подання заявок на купівлю та продаж, мета ринкового оператора полягає у пошуку рішення, яке визначає 24 ціни клірингу (market clearing price) відповідно до 24-годинного графіка планування і присвоєння задач на генерацію енергії кожному учаснику-продавцю енергії, що приймав участь у торгах.

Алгоритм роботи ринкового оператора для визначення пар продавця-покупця описується наступними кроками:

— розрахунок точки перетину між сукупними кривими попиту та пропозиції та розрахунок відповідної погодинної ціни клірингу; кожна клірингова

ціна відповідає пропозиції на продаж останньої виробничої одиниці, що була необхідною для задоволення загального попиту за цей період;

- присвоєння кожному продавцю завдання на генерацію тієї кількості енергії, що була зазначена у ставці, за умови, що ціна у пропозиції на продаж є нижчою або дорівнює граничній ціні та наявності достатнього попиту на енергію за цією ціною або вище;

- присвоєння кожному покупцеві визначеної у ставці кількості енергії за умови, що ціна купівлі є вищою або дорівнює граничній ціні та наявності достатнього попиту на енергію за цією ціною або нижче;

- оператор перевіряє чи всі прийняті угоди відповідають обмеженням на швидкість збільшення та зменшення продукції; якщо учасник ринку виробляє кількість продукції, що перевищує обмеження, ця кількість зменшується у відповідності до швидкості збільшення продукції і навпаки;

- оператор перевіряє умову нероздільної кількості продукції: невідповідні ставки впорядковуються за збільшенням кількості продукції та виводяться по черзі, починаючи з найменшої, поки решта не стануть відповідати умові;

- оператор перевіряє виконання умови мінімального доходу для кожного учасника: вони упорядковуються відповідно до різниці між фактичним та мінімальним доходом таким чином, що учасник із найбільшою різницею першим виходить з аукціону.

Описаний алгоритм покладено в основу багатьох варіацій алгоритмів управління в DEMS.

Відмінність використаного у розробленій системі алгоритму від типового представника сімейства Day-Ahead Market полягає у відмінностях проектування системи енергоменеджменту. Виходячи із застосування гібридної архітектури для побудови мультиагентної моделі мікроенергостанції, алгоритм енергоринку також було модифіковано. Очищення ринку в застосованому алгоритмі відбувається не за рахунок визначення ціни клірингу, а за рахунок визначення оптимальних цін та обсягів генерації для кожного учасника-продавця шляхом вирішення задачі про

оптимальний потік енергії в системі.

Детальний опис особливостей архітектури розробленої системи та використаного алгоритму наведено у наступному розділі.

Висновки до розділу 2

Визначено переваги агентного підходу для моделювання режимів функціонування технологічних систем.

В результаті аналізу та порівняння доступних платформ мультиагентного моделювання обрано фреймворк Repast для програмної реалізації визначеного задачею продукту.

В результаті аналізу підходів до побудови інформаційних систем управління мікроенергостанції обрано гібридну децентралізовану модель архітектури та енергоринку як засіб організації управління. У якості алгоритму забезпечення функціонування системи обрано аукціон закритого типу першої ціни — модифікацію алгоритму Day-Ahead Market.

Визначено інноваційну новизну продукту, що полягає у вдосконаленні існуючих систем моделювання та управління роботою мікроенергостанцій за рахунок:

- застосування мультиагентного підходу для формування та управління енергетичним аукціоном і, відповідно, управління енергетичною інфраструктурою;

- використання уніфікованої моделі генератора, що з одного боку спрощує розробку та масштабування мультиагентної системи, а з іншого боку враховує індивідуальні особливості генератора (вітряк, СЕС тощо) через його енергетичні характеристики;

- впровадження стратегії поведінки учасників торгів у умовах енергоринку.

3. ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ МОДЕЛЮВАННЯ МІКРОЕНЕРГОСТАНЦІЙ

Після аналізу поставленої задачі та виділення критеріїв, яким має задовольняти розроблювана система, було визначено основні та додаткові інструменти для реалізації програмного агента моніторингу та управління режимами роботи мікроенергостанцій та, відповідно, створення середовища для його випробування, тобто системи моделювання мікроенергостанцій.

Зважаючи на наявність відповідних технологій та бібліотек, у якості мови розробки було обрано мову Java, а у якості середовища розробки — продукт IntelliJ Idea 2019 та модуль його розширення для зручної побудови графічних інтерфейсів JFormDesigner.

Для побудови мультиагентної системи із можливістю симуляції, відповідно до результатів порівняння двох основних платформ мультиагентного моделювання, наведених у попередньому розділі, обрано бібліотеку Repast та її конкретну реалізацію для мови Java — RepastJ.

Для вирішення задачі оптимізації використано бібліотеки QuadProg та DCOPFJ. Бібліотека Repast також надає можливості розширення функціоналу, зокрема для побудови інтелектуальних агентів із можливістю навчання. Для навчання агентів використано модуль розширення бібліотеки Repast — JReLM. Графічний інтерфейс користувача побудовано із використанням технології Swing.

3.1 Програмна модель мікроенергостанції

Першим кроком розробки після визначення задачі та вимог до функціоналу стало визначення того, яким чином фізична модель мікроенергостанції або мікроенергомережі має бути відображена у програмній реалізації. Відповідно, постало питання: які елементи мікроенергостанції є ключовими для створення

адекватного середовища випробування розроблюваного агента, а які можна відкинути. В результаті ознайомлення із необхідними вхідними параметрами задачі про оптимальний потік енергії у мікроенергостанції, яку повинен вирішувати програмний агент моніторингу та управління режимами її роботи було обрано модель, зображену на рисунку 3.1.

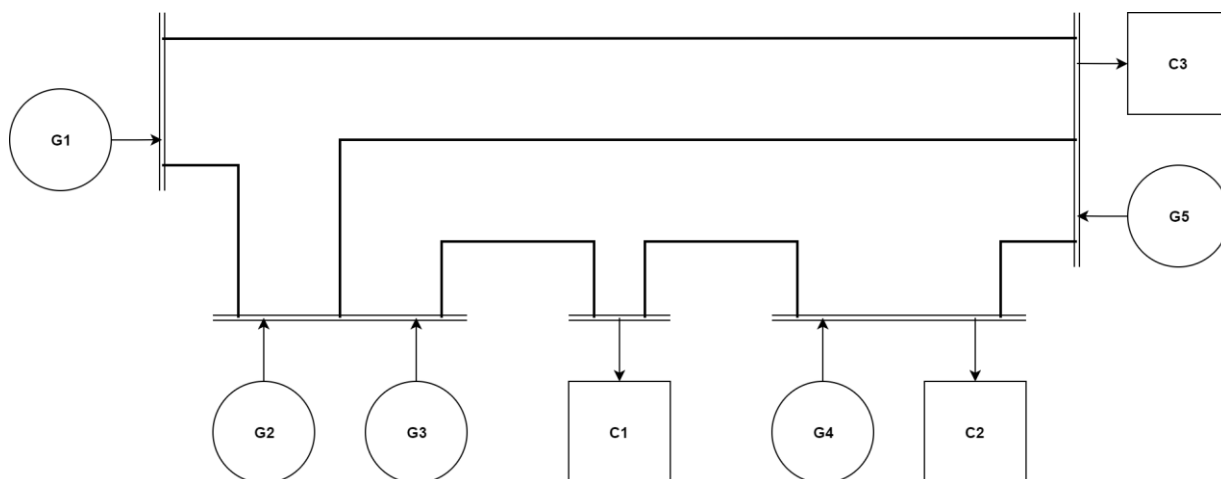


Рисунок 3.1 — Схема моделі мікроенергостанції

Відповідно до представленої схеми, у якості ключових елементів мікроенергостанції, що обов'язково мають бути відображені у програмній моделі, обрано наступні:

- генератори та їх характеристики, пов'язані із розташуванням у мережі, можливим обсягом вироблення енергії та вартістю експлуатації;
- споживачі, що як і генератори характеризуються своїм розташуванням, а також обсягом споживання, що визначається погодинним графіком навантаження;
- вузли та їх фізичні характеристики, а також розташування у мережі;
- лінії передач та їх фізичні характеристики (опір, пропускна здатність тощо).

У запропонованій моделі мікроенергостанції агентами виступають генератори та споживачі. Розроблюваний агент-координатор є частиною інформаційного шару мережі, не має відповідної фізичної сутності і не приймає участі у обміні енергоресурсами, тому не є частиною фізичної моделі.

3.2 Загальний опис системи

Для моделювання мікроенергостанції та симуляції її роботи застосовано мультиагентне моделювання енергоринку. Енергоринок за своїм принципом роботи аналогічний ринку, що застосовується у глобальних енергетичних мережах, тобто є варіацією алгоритму Day-Ahead Market.

Змодельована система працює на визначеній користувачем мережі енергопередачі, починаючи з 1-го дня та продовжується до визначеного користувачем фінального дня симуляції (якщо не припиняється раніше відповідно до визначеного користувачем правила зупинки).

Кожен день D складається з 24 послідовних годин $H = 00, 01, \dots, 23$.

Ринок електроенергії включає розроблюваного агента-координатора та групу учасників ринку, яка складається із споживачів та генераторів, розподілених по електромережі та об'єднаних змодельованими лініями електропередачі. На рисунку 3.2 зображена загальна схема роботи енергоринку розробленої системи.

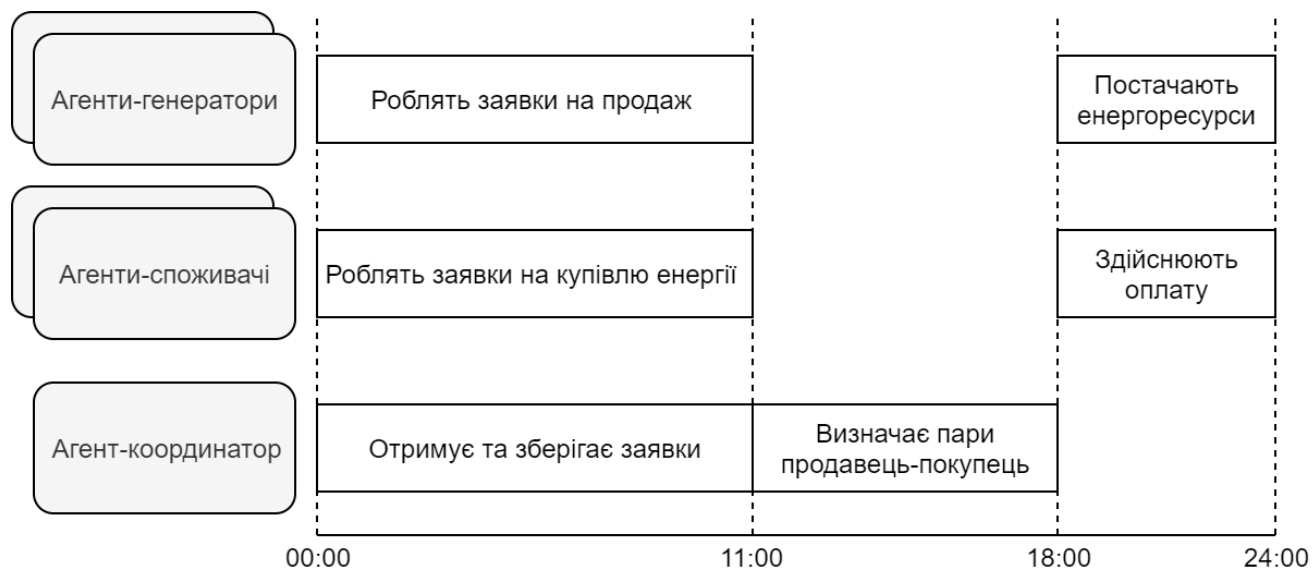


Рисунок 3.2 — Схема роботи створеного енергоринку

Завданням координатора є досягнення експлуатаційної ефективності мікроенергостанції за рахунок утвореного енергоринку з урахуванням обмежень на генерацію та транспортування енергії. Намагаючись досягти цієї мети, координатор забезпечує щоденне функціонування ринку шляхом визначення локального граничного ціноутворення (LMP), тобто визначення цін на енергію відповідно до розміщення учасників ринку та термінів її введення або виведення з енергомережі.

Завдання кожного агента-споживача — забезпечити енергоспоживання для своїх користувачів. Це може бути як один будинок, так і деяка кількість будинків, що підключені до апаратної частини агента. Вранці кожного дня D кожен агент-споживач надсилає заявку на енергію до координатора для ринку $D+1$, тобто на наступний день.

Кожна така заявка складається з двох частин: фіксованої пропозиції на купівлю (цілодобового профілю споживання), і 24-х ставок, що враховують ціну на енергію (по одній на кожну годину). Кожна чутлива до ціни ставка складається з лінійної функції, визначеної на інтервалі спроможності покупки. Графік споживання енергії та чутливі до ціни ставки визначаються користувачем та встановлюються на початку кожного запуску моделювання.

Завдання кожного генератора — забезпечити для себе максимально можливий чистий прибуток щодня. Вранці кожного дня D кожен генератор використовує поточні ймовірності вибору дій, щоб вибрати пропозицію із простору дій (Action Domain), щоб повідомити координатор про бажані режими роботи протягом усіх 24 годин ринку наступного дня.

Кожна пропозиція просторі дій генератора складається з лінійної функції граничної вартості, визначеної на інтервалі робочої потужності. Можливість генератора змінювати свій вибір пропозиції дозволяє йому коригувати нахили своєї функції граничної вартості, яка повідомляється агенту-координатору, та верхню межу звітного інтервалу робочої потужності, з метою збільшення свого чистого щоденного прибутку.

Після отримання заявок на попит від споживачів та пропозицій від генераторів протягом ранку дня D координатор визначає та повідомляє усім агентам учасникам ринку їх погодинні зобов'язання щодо постачання електроенергії та локальні граничні ціни для ринку на день $D+1$, як вирішення задачі про оптимальний потік енергії.

Управління навантаженням на лінії передачі здійснюється за рахунок включення компонентів витрат на перевантаження у локальні ціни, відповідно до розташування агента у мережі.

Наприкінці кожного дня D координатор узгоджує всі зобов'язання щодо ринку напередодні на день $D+1$.

Наприкінці кожного дня D кожен генератор використовує модуль навчання з підкріпленням для оновлення ймовірностей вибору дій, із врахуванням отриманого чистого доходу за день D .

3.3 Опис архітектури системи

Зважаючи на описані у попередніх розділах недоліки централізованих та децентралізованих систем, було прийнято рішення створювати мультиагентну систему енергоменеджменту у відповідності до гібридної архітектури.

Ієрархічна архітектура мікроенергостанції має багато спільних рис із типовою децентралізованою архітектурою з єдиною відмінністю, що полягає у присвоєнні одному із агентів дещо більшого рівня відповідальності. Таким чином, розроблений агент моніторингу та управління режимами роботи мікроенергостанцій виступає у ролі координатора системи та оператора енергоринку. Він забезпечує стабільність функціонування системи, але не є її центральним контролером. Інші агенти системи, що представлені агентами-генераторами та агентами-споживачами зберігають свою автономність та цілісність. Вони запрограмовані на досягнення власних цілей, що враховується агентом-координатором.

3.2.1 Огляд архітектури

Основними чітко розмежованими компонентами розробленої системи є ядро програми та модуль графічний інтерфейсу.

Програмний продукт було створено із використанням мови Java, тож компоненти системи було розбито на пакети. Відповідно, система складається із головного пакету (Core) та пакету із графічним користувацьким інтерфейсом (GUI).

Головний пакет вміщає ядро системи, що представлене наступними підпакетами: мультиагентна модель мікроенергостанції, модуль управління симуляцією, обчислювальні компоненти та утилітарні модулі: модуль роботи з файлами, валідації даних тощо.

Головним модулем ядра системи є мультиагентна модель мікроенергостанції, а саме реалізація енергоринку. Модуль представлений у пакеті market наступними класами (рисунок 3.3):

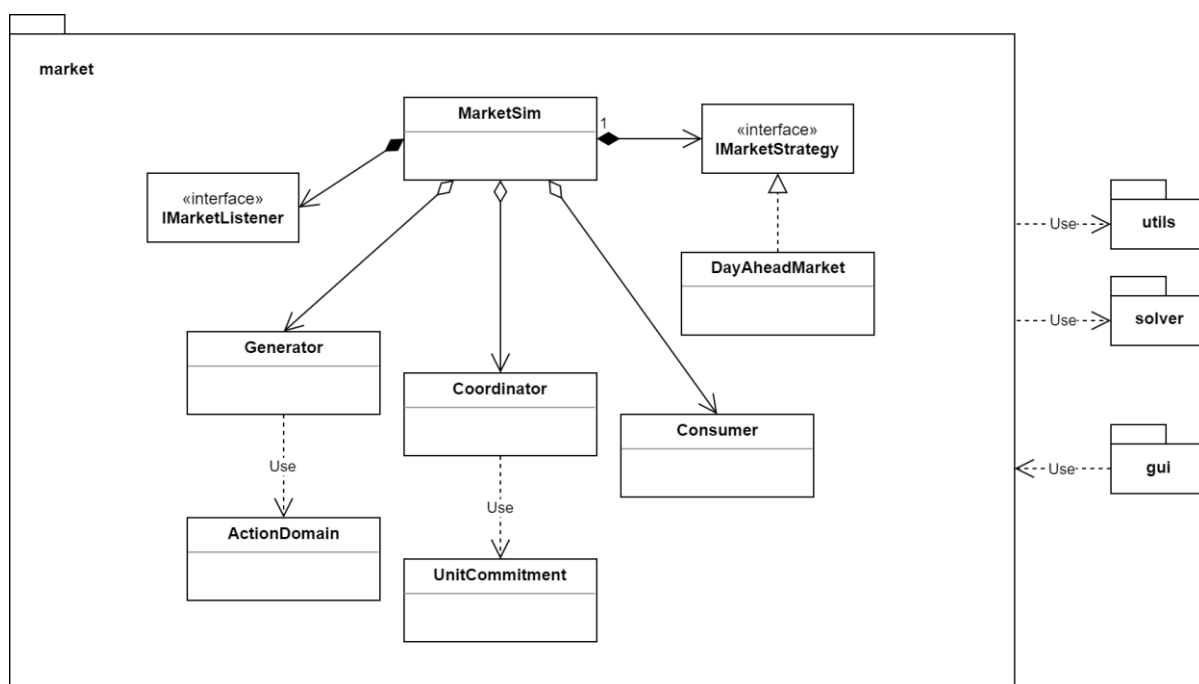


Рисунок 3.3 — Діаграма класів пакету market

— MarketSim — клас, що наслідує модель бібліотеки Repast, є ключовим

для налаштування та симуляції системи; клас відповідає за життєвий цикл агентів у складі модельованої мультиагентної системи;

- `DayAheadMarket` — клас, який є конкретною реалізацією інтерфейсу `IMarketStrategy`; для забезпечення можливостей зручного розширення системи у майбутньому, `MarketSim` працює за алгоритмом, який отримує через інтерфейс `IMarketStrategy`, тобто у пакеті `market` реалізовано патерн проектування Стратегія;

- класи `Generator`, `Consumer` та `Coordinator` представляють відповідні програмні агенти;

- `ActionDomain` — клас, що моделює середовище вибору можливих дій (AD) для агента-генератора, клас необхідний для реалізації навчання з підкріпленням, що описане у одному із наступних пунктів розділу;

- `UnitCommitment` — клас, який представляє собою графік режимів роботи агентів системи; екземпляри класу створюються та передаються агенту-координатору у результаті вирішення задачі про оптимальний потік енергії в системи засобами, описаними у пакеті `solver`; клас дозволяє зручно отримувати доступ до даних, які мають бути запаковані у `Repast`-повідомлення та надіслані відповідним агентам;

- інтерфейс `IMarketListener` описує протокол комунікації ядра програми (мультиагентної моделі) із іншими компонентами системи, для отримання сповіщень про зміни в системі та обміну даними, уникаючи сильної зв'язності в системі.

Пакет `market` не може існувати без пакету `solver` (рисунок 3.4), який використовується агентом-координатором (`Coordinator`) для забезпечення стабільного функціонування енергоринку та оптимальної зміни режимів роботи мікроенергостанції.

Клас `QuadProg` є класом відповідної бібліотеки для вирішення задач квадратичного програмування. `QuadProg` не розрахована на застосування у деякій конкретній предметній області і надає загальні можливості для вирішення оптимізаційних задач.

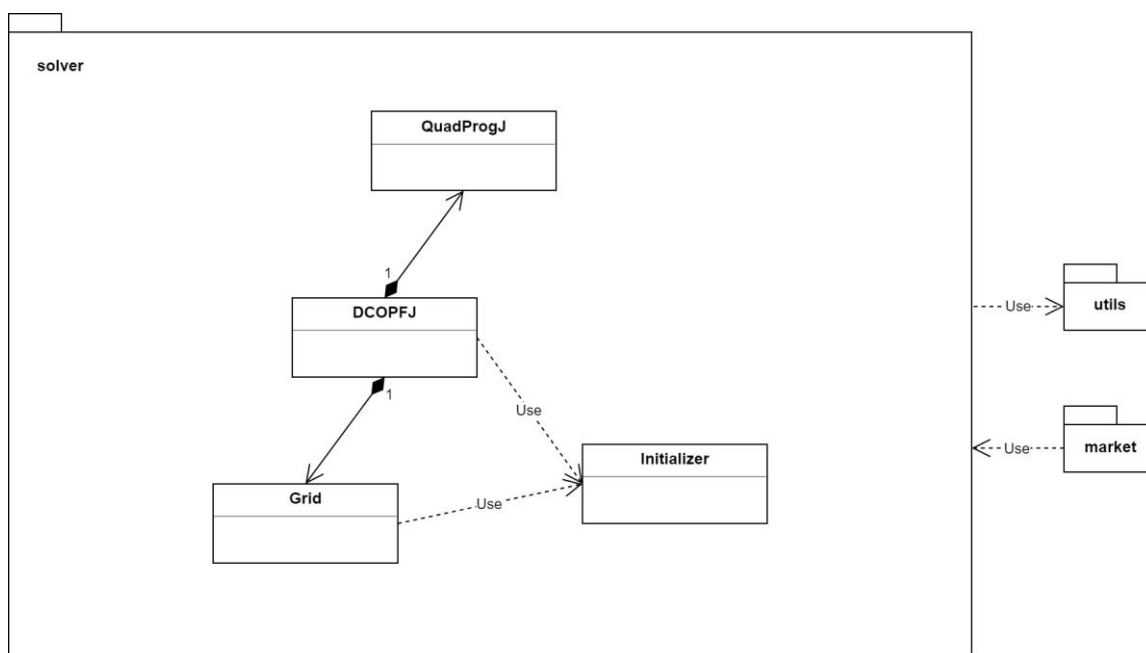


Рисунок 3.4 — Діаграма класів пакету solver

Виходячи з цього, було використано DCOPFJ — оболонку над QuadProg, що використовує дані про конфігурацію та поточний стан енергетичної системи для формування вхідних даних у формі, що очікує QuadProg.

Клас Grid представляє собою структуру збереження даних про мережу мікроенергостанції: шини та лінії електропередачі, їх розміщення, фізичні властивості та агентів системи, яких вони об’єднують. Ця інформація необхідна для вирішення задачі про оптимальний потік енергії у мережі та передається у QuadProg через оболонку DCOPFJ.

Клас Initializer використовується для заповнення екземплярів класів Grid та DCOPFJ початковими даними. Він виконує утилітарну функцію: так як запис великої кількості даних у будь-якому іншому місці, наприклад MarketSim, призвів би до порушення принципу single responsibility.

Класи обох пакетів також використовують пакет Utils, що містить статичні класи, які надають різноманітні утилітарні методи для зчитування, форматування та запису даних, визначення параметрів за замовчуванням тощо.

Пакет користувацького інтерфейсу містить компоненти, що надають користувачу системою можливість зручно використовувати її функціонал.

3.2.2 Конфігурація програмних агентів

Агенти-споживачі закупають енергію оптом кожного дня із метою забезпечити свої потреби. Користувач системи визначає їх кількість, а також їх розміщення на вузлах (шинах) мережі мікроенергостанції. Агенти-споживачі не можуть приймати участь у ринку у якості постачальників енергії, тобто агенти-споживачі можуть купляти енергію лише у агентів-генераторів, а не один у одного. Агенти-споживачі є пасивними, у тому сенсі, що вони подають заявки на покупку енергії виключно у відповідності до свого графіку споживання, не передбачаючи їх стратегічний розгляд або прогнозування найбільш вигідного графіка навантажень. Цей функціонал може бути впроваджений у майбутньому.

Агенти-генератори представляють собою розподілену систему енергогенерації мікроенергомережі. Як і у випадку агентів-споживачів, користувач системи визначає їх кількість та розміщення на вузлах мережі мікроенергостанції. Агенти-генератори можуть продавати енергію лише агентам-споживачам. Кожен генератор конфігурується параметрами видобутку енергії, параметрами модуля машинного навчання та початковою кількістю фінансових ресурсів. Генератори мають фіксовану та змінну вартість експлуатації. Вартість запуску та зупинки генератора не береться до уваги у розрахунках, як і обмеження на різке збільшення або зменшення продукції.

Агент-координатор фізично не присутній у мережі мікроенергостанції, він є виключно програмною сутністю її інформаційного шару. Задача агента-координатора є схожою на задачу ISO (Independent System Operator) у глобальних енергетичних мережах. Обов'язком агента-координатора є визначення графіку оптимальних режимів роботи агентів системи та локальних граничних цін для кожної з 24-х годин робочого дня. Графік базується на попиті, пропозиції, теплових обмеженнях на потоки енергії у мережі та обмеженнях на енергетичний баланс на кожному з вузлів мережі. Оптимальні режими роботи — такі, що мінімізують загальні чисті витрати, що відповідають попиту, сформованому виходячи із цін на енергію.

3.2.3 Модель даних

Для збереження даних про конфігурацію мультиагентної моделі, а також для збереження результатів симуляції було розроблено спеціальні формати документів, що є зручними як для перегляду у текстових редакторах, так і для зчитування засобами розробленого програмного додатку.

Файли конфігурації мають розширення *.dat та форматуються згідно із описаними далі правилами.

Перша графа файлу має вигляд // FILENAME <назва файлу>. Наступна, обов'язкова для коректного зчитування даних, графа має вигляд // UNIT SI. Після неї вказуються одиниці виміру даних, відповідно до системи SI. Далі файл поділяється на категорії виду #CategoryStart <дані категорії> #CategoryEnd. Типовими категоріями для файлу вхідних даних конфігурації мультиагентної моделі є кількість вузлів (шин) у мережі (NodeData), дані та параметри ліній передач (BrachData), параметри агентів-генераторів (GenData), профілі споживання агентів-споживачів, що представлені у вигляді фіксованого попиту на енергію (LSEDataFixedDemand), графік чутливого до ціни попиту на енергію (LSEDataPriceSensitiveDemand), параметри модуля машинного навчання для агентів-генераторів (GenLearningData).

Вихідні файли із результатами, отриманими під час симуляції, мають розширення *.out та містять обов'язкові графи із зазначенням назви файлу та конфігурації. Файли результатів також поділені на категорії, що відповідають отриманим результатам симуляції системи.

3.3 Задача про оптимальний потік енергії у мережі

Як було визначено у попередніх пунктах розділу, розроблений програмний агент забезпечує автоматизовану зміну режимів роботи мікроенергомережі, вирішуючи задачу про оптимальний потік енергії у мережі на кожній ітерації ринку, що працює у відповідності із алгоритмом DAM.

Далі наведено математичне формулювання задачі, згідно з проаналізованою літературою [36].

Кожен агент-генератор i має функцію загальної експлуатаційної вартості:

$$TC_i(p_{Gi}) = a_i \cdot p_{Gi} + b_i \cdot p_{Gi}^2 + FCost_i$$

де p_{Gi} — обсяг генерації генератора G_i ;

$FCost_i$ — фіксована вартість генератора G_i .

Звідки, функція змінної вартості генератора i :

$$TVC_i(p_{Gi}) = a_i \cdot p_{Gi} + b_i \cdot p_{Gi}^2$$

Функція граничної вартості генератора i (функція пропозиції):

$$MC_i(p_{Gi}) = a_i + 2 \cdot b_i \cdot p_{Gi}$$

Запит на потребу в енергії агента-споживача j складається з двох частин: фіксованої ставки p_{Lj}^F та змінної ставки p_{Lj}^S , тобто:

$$p_{Lj} = p_{Lj}^F + p_{Lj}^S$$

Функція чутливого до ціни запита на потребу в енергії агента-споживача j , що визначає максимальне бажання платити як функцію від бажаної кількості p_{Lj}^S :

$$D_j(p_{Lj}^S) = c_j - 2 \cdot d_j \cdot p_{Lj}^S$$

Валовий надлишок агента-споживача j , відповідно до його чутливого до ціни запита на потребу в енергії:

$$GSS_j(p_{Lj}^S) = c_j \cdot p_{Lj}^S - d_j \cdot p_{Lj}^{S^2}$$

Загальний чистий надлишок, відповідно до чутливого до ціни запиту:

$$TNSS(p_G, p_L^S) = GSS(p_L^S) - TVC p_G$$

де $p_G = (p_{G1}, p_{G2}, \dots, p_{GI})$;

$p_L^S = (p_{L1}^S, p_{L2}^S, \dots, p_{LJ}^S)$;

$GSS(p_L^S) = \sum_{j=1}^J GSS_j(p_L^S)$;

$TVC(p_G) = \sum_{i=1}^I TVC_i(p_{Gi})$.

Функція загальної чистої вартості, відповідно до чутливих до ціни запитів:

$$TNSC(p_G, p_L^S) = -TNSS(p_G, p_L^S)$$

Поширене у літературі представлення задачі про погодинне знаходження оптимального потоку енергії — мінімізація загальних чистих витрат, що відповідають чутливому до цін попиту $TNSC$ із врахуванням обмежень щодо енергопередачі, тобто:

$$\min TNSC(p_G, p_L^S) + \pi \left[\sum_{km \in BR} [\delta_k - \delta_m]^2 \right]$$

за умов:

$$p_{Gi}, i = 1, \dots, I; p_{Lj}^S, j = 1, \dots, J; \delta_k, k = 1, \dots, K;$$

відповідно до обмеження енергетичного балансу для кожного вузла $k = 1, \dots, K$:

$$\sum_{i \in I_k} p_{Gi} - \sum_{j \in J_k} p_{Lj} - \sum_{km \text{ or } mk \in BR} p_{km} = 0$$

де $p_{Lj} = p_{Lj}^F + p_{Lj}^S$;

$$P_{km} = B_{km}[\delta_k - \delta_m];$$

теплого обмеження для кожної лінії передач $km \in BR$:

$$|P_{km}| \leq P_{km}^U$$

обмеження робочої потужності для кожного агента-генератора i :

$$Cap_i^L \leq p_{Gi} \leq Cap_i^U$$

діапазону допустимих цін кожного агента-споживача j :

$$SLoad_j^L \leq p_{Lj}^S \leq SLoad_j^U$$

та налаштування кута напруги на вузлі 1:

$$\delta_1 = 1$$

Для вирішення задачі у розробленій системі застосовано розв'язувач QuadProg. QuadProg — це автономний розв'язувач на мові Java з відкритим кодом для строго випуклих задач квадратичного програмування. QuadProg реалізує відомий подвійний алгоритм вирішення задач квадратичного програмування з подвійною активною множиною, розроблений Goldfarb та Idnani.

QuadProg успішно пройшов перевірку на малих та середніх задачах квадратичного програмування. QuadProg досягав відповідної або перевищував точність QR-розв'язувача BPMPD на мові C при тестуванні, що проводилось на базі загальнодоступного сховища строго випуклих задач квадратичного програмування із близько 1500 змінних рішень та обмежень [37]. BPMPD є надійним пакетом, що використовується з 1998 року та демонструє виняткову ефективність. Таким чином, QuadProg можна вважати одним із найбільш надійних серед доступних інструментів розв'язання ЗКП.

Пакет DCOPFJ використовується для вирішення задачі оптимального потоку електроенергії мікроенергомережі. DCOPFJ — автономний розв'язувач

DC-OPF на мові Java, що складається із ядра QuadProg, яке загорнуто в оболонку для обробки даних.

Оболонка обробки даних послідовно виконує наступні дії: перетворення всіх вхідних даних із SI (міжнародна система одиниць) у нормалізовану форму PU (Per Unit), формування всіх необхідних матриць і векторів для вираження розширеної задачі у матричній формі, виклик QuadProg для визначення рішення результуючої матриці та перетворення всіх вихідних даних рішення з PU у форму SI [38].

3.4 Навчання програмних агентів

Для більш точного відображення реального енергоринку та забезпечення гібридної архітектури системи енергоменеджменту для розробленої моделі мікроенергостанції, вирішено вдосконалити агенти-генератори можливостями машинного навчання.

Кожен агент-генератор адаптивно обирає пропозиції на продаж енергії, виходячи із своїх власних минулих доходів, використовуючи варіацію стохастичного навчання з підкріпленням, запропонованого Roth-Erev. Алгоритм засновано на психологічних експериментах, проведених над людьми. У літературі він також відомий як VRE Learning Algorithm.

3.4.1 Алгоритм VRE

Відмінність алгоритму навчання з підкріпленням від алгоритму, запропонованого дослідниками Sutton та Barto полягає у відмові від поняття стану. Тобто, згідно з алгоритмом стан навколишнього середовища, в якому навчається агент, не розглядається, а натомість робота алгоритму базується на останній обраній агентом дії та нагороді, отриманій в наслідок цієї дії. Також алгоритм не описує жодної чіткої політики або функції визначення вибору дій, але зберігає ймовірнісний розподіл.

Базовий RE алгоритм полягає у присвоєнні значень схильності усім можливим діям у просторі вибору дій (Action Domain). Із цих схильностей обчислюється ймовірнісний розподіл, яким урегульовується майбутній вибір дій.

Позначимо $q_j(t)$ схильність для вибору дії j у момент часу t , де j — одна з n дій, описаних у просторі вибору дій, а t — одиниця виміру циклів навчання. Тоді ймовірність вибору j у момент t :

$$p_j(t) = \frac{q_j(t)}{\sum_{m=1}^n q_m(t)}$$

У момент ініціалізації до входу у перший цикл навчання, усі схильності вибору дій ініціалізуються з однаковим значенням схильності q_{init} , таким чином початковий ймовірнісний розподіл буде рівномірним.

Ядром базового RE алгоритму є функція підкріплення, яка відповідає за оновлення схильності вибору дій. При виборі агентом дій, схильності оновлюються за допомогою отриманих відгуків (винагород). Коли агент вибір дії k у момент часу t і отримує винагороду $r_k(t)$, схильність вибору дії j у момент часу $t+1$ оновлюється у відповідності до:

$$q_j = \begin{cases} q_j(t) + r_k(t) & \text{якщо } j = k \\ q_j(t) & \text{якщо } j \neq k \end{cases}$$

Таким чином, схильність вибору дії k посилюється позитивно чи негативно залежно від отриманої винагороди. Схильності для всіх інших дій залишаються незмінними.

Було також запропоновано оновлену версію алгоритму із альтернативною функцією підкріплення:

$$q_j(t+1) = [1 - \phi]q_j(t) + E_j(\epsilon, j, k, t)$$

де ϕ — параметр новизни.

Параметр новизни визначає наскільки недавнє минуле впливає на поточний вибір дії, порівняно із далеким минулим. Тобто параметр визначає швидкість з якою винагороди минулого стираються з пам'яті агента. Це виконується шляхом поступового зменшення старіших схильностей під час кожного їх оновлення. Параметр новизни іноді називають параметром забуття.

Зі збільшенням φ нова схильність дії j , $q_j(t+1)$ стає меншою за своє попереднє значення $q_j(t)$. Значення схильності, накопичене на основі попередніх винагород, має слабший вплив на нове значення. Значення схильності, що згасає, доповнюється значенням досвіду, заданим функцією досвіду:

$$E_j(\epsilon, j, k, t) = \begin{cases} r_k(t)[1 - \epsilon] & \text{якщо } j = k \\ r_k(t)\frac{\epsilon}{2} & \text{якщо } j = k \pm 1 \end{cases}$$

де ϵ — параметр експериментування.

Параметр експериментування впливає на те, з якою ймовірністю агент повинен обирати дії, що схожі на дії, що вже були обрані у минулому.

У випадку, коли вибір дій у просторі вибору не впорядковано відповідно до їх подібності, функція досвіду має наступний вигляд:

$$E_j(\epsilon, j, k, t) = \begin{cases} r_k(t)[1 - \epsilon] & \text{якщо } j = k \\ r_k(t)\frac{\epsilon}{N-1} & \text{якщо } j \neq k \end{cases}$$

де N — загальна кількість дій, що можуть бути обрані агентом.

Базовий RE алгоритм, однак, страждає від проблеми, яка може перешкодити навчальному процесу у контексті, коли агент може отримувати нульові винагороди. Коли агент отримує нульову винагороду обидві функції досвіду не призводять до зміни ймовірностей вибору дій. Для уникнення цієї проблеми було запропоновано модифікацію алгоритма — VRE. Відповідно, нова функція досвіду має вигляд [39]:

$$E_j(\epsilon, k, t) = \begin{cases} r_k(t)[1 - \epsilon] & \text{якщо } j = k \\ q_j(t) \frac{\epsilon}{N-1} & \text{якщо } j \neq k \end{cases}$$

На початку кожного дня D генератор i має обрати пропозицію із свого простору дій (AD) для повідомлення агента-координатора про участь у ринку дня $D+1$. Початкова схильність вибору пропозиції $m \in AD_i$ позначена $q_{im}(0)$. Початкові схильності однакові для усіх генераторів. Відповідно, ймовірності вибору дій відповідно до схильностей визначаються за формулою:

$$p_{im}(D) = \frac{\exp(q_{im}(D)/C_i)}{\sum_{j=1}^{M_i} \exp(q_{ij}(D)/C_i)}, m \in AD_i$$

де C_i — параметр охолодження.

Параметр охолодження визначає рівень впливу значень схильності на визначення ймовірностей вибору.

Наприкінці кожного дня D поточна схильність $q_{im}(D)$, що відповідає кожній пропозиції $m \in AD_i$ оновлюється відповідно до правила:

$$q_{im}(D + 1) = [1 - r_i]q_{im}(D) + Response_{im}(D)$$

де

$$Response_{im}(D) = \begin{cases} [1 - e_i] \cdot Profit_{im'}(D) & \text{якщо } m = m' \\ e_i \cdot q_{im}(D)/[M_i - 1] & \text{якщо } m \neq m' \end{cases}$$

3.4.2 Засоби реалізації

Для реалізації VRE алгоритму у розробленій системі застосовано бібліотеку JReLM. Java Reinforcement Learning Module (JReLM) — це платформа для розробки та використання навчання з підкріпленням агентів у рамках моделювання Repast.

Компоненти JReLM частково мотивовані описом Саттона та Барто агента з підкріпленням. Однак JReLM зосереджена на побудові підрозділів, які навчаються на досвіді агента, а потім керують поведінкою агента на основі цього досвіду, а не на побудові самих агентів. Конкретні деталі агентів можуть бути адаптовані до індивідуального моделювання, залишаючи реалізацію навчання з підкріплення JReLM. JReLM покликаний забезпечити простий для програміста спосіб впровадження навчання з підкріпленням в мультиагентні системи, створені на базі платформи моделювання Repast.

3.5 Графічний інтерфейс користувача

Важливою вимогою конкурентоспроможності програмного продукту є зручний графічний користувацький інтерфейс, що надає користувачам програми доступ до її функціоналу.

Для розробленої системи моделювання та симуляції мультиагентної моделі мікроенергостанції було розроблено графічний інтерфейс із використанням технології Swing. У наступних підпунктах наведено короткий огляд технології, розглянуто та проілюстровано доступний користувачам функціонал.

Елементи графічного інтерфейсу програми представлені у пакеті gui файлами із розширенням *.jfd (розширення інструменту побудови Swing-форм JFormDesigner). Кожному jfd-файлу поставлено у відповідність Java клас, що відіграє роль контролера та пов'язує елементи інтерфейсу із відповідними сутностями у моделі.

На рисунку 3.5 наведено головне вікно програми. Даний елемент інтерфейсу є вхідною точкою програми. Саме в ньому користувач має змогу виконувати всі необхідні дії та отримувати інформацію. Головному вікну програми відповідає контролер Main.

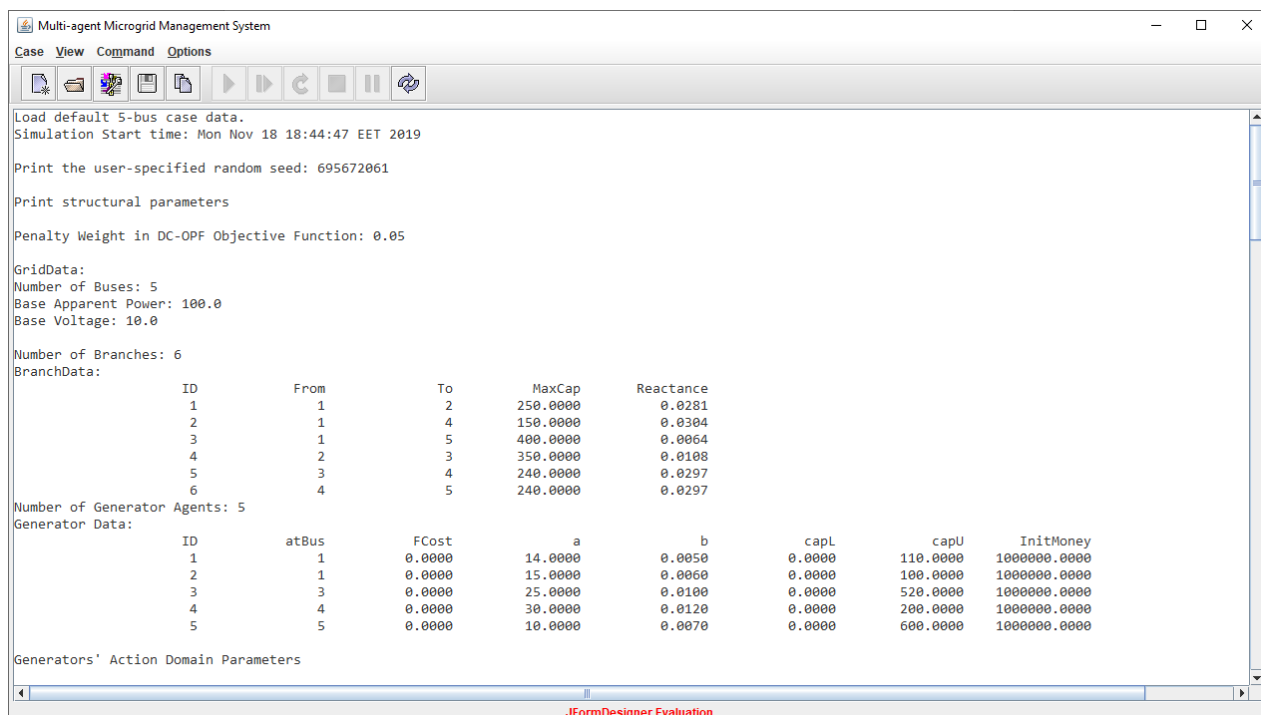


Рисунок 3.5 — Головне вікно програми

Більшу частину інтерфейсу головного вікна займає панель виводу даних про симуляцію. Сюди у зручній для перегляду формі виводяться дані, зчитані програмою із файлу конфігурації моделі, налаштування користувача та загальні відомості про хід симуляції.

Користувачу доступні пункти меню Case — для створення, збереження відкриття існуючих та налаштування конфігурацій модельованої мікроенергостанції, View — для перегляду отриманих даних у вигляді таблиць або графіків, що детально описано далі, Command — для управління симуляцією та Options — для налаштування параметрів машинного навчання агентів-генераторів та параметрів симуляції.

Найбільш використовуваний функціонал винесено на панель інструментів головного вікна програми.

Процес налаштування моделі мікроенергостанції розбито на декілька кроків. Перший крок — налаштування глобальних параметрів (рисунок 3.7).

Step 1: Input global parameters

Input total number of

Grid Buses	5
Grid Branches	6
Generators	5
Consumers	3

JFormDesigner Evaluation

Set base values for voltage and apparent power

Voltage Base	10.0
Apparent Power Base	100.0

Cancel Next >>

Рисунок 3.7 — Вікно налаштування глобальних параметрів

Наступними кроками є налаштування параметрів мережі, параметрів агентів-генераторів, параметрів навчання агентів генераторів (рисунок 3.8), параметрів агентів-споживачів та параметрів симуляції.

Step 4: Variant of Roth Erev RL: Parameters

Generators: GenCo1

Reward Selection

☐ Profits ☒ Net Earnings

Learning Parameters

Initial Propensity:	6000.0
Cooling:	1000.0
Recency:	0.04
Experimentation:	0.96

JFormDesigner Evaluation

Action Domain Parameters

M1:	10
M2:	10
M3:	1
RIMaxL:	0.75
RIMaxU:	0.75
RIMinC:	1.0
Slope Start:	0.001

Cancel << Prev Next >>

Рисунок 3.8 — Вікно налаштування параметрів навчання генераторів

Користувач системи має змогу задати кількість днів, протягом яких буде відбуватись симуляція роботи мережі, граничну ймовірність вибору певної дії

(пропозиції) агентом-генератором, додаткові параметри перевірки ймовірності, стабільності та максимального валового доходу за день симуляції для агентів-генераторів.

Важливою функцією розробленої системи є можливість представлення результатів симуляції у вигляді таблиць та графіків (рисунок 3.10).

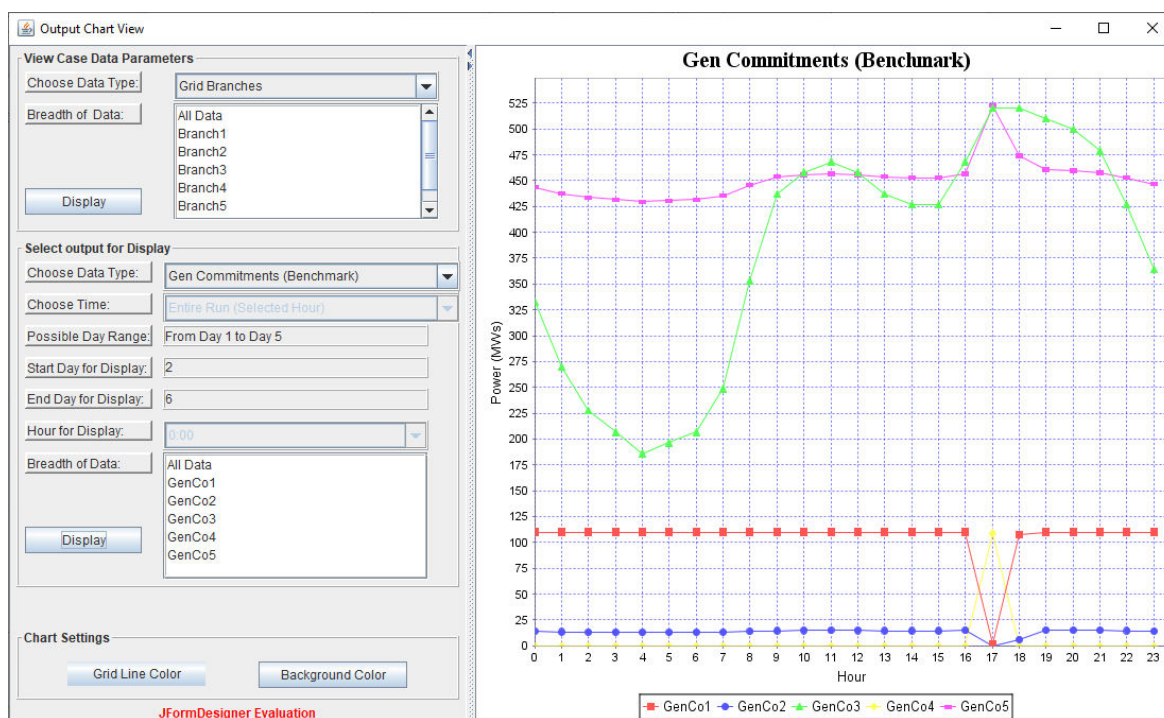


Рисунок 3.10 — Вікно відображення результатів у вигляді графіків

Окремим налаштуванням у даному вікні є параметри графічного відображення. Для зручності користувач може змінити колір ліній графіків, налаштувати написи та колір фону.

Висновки до розділу 3

Розроблено програмну систему мультиагентного моделювання мікроенергостанцій та агент моніторингу та управління режимами роботи мікроенергостанцій.

Завдяки застосуванню мультиагентного підходу, розроблена система дозволяє симулювати роботу мікроенергостанції для дослідження функціонування агента моніторингу та управління режимами роботи у її складі. Мультиагентний підхід також забезпечує наближеність програмної моделі мікроенергостанції до її фізичного представлення. Таким чином, програмна модель може бути використана для проектування реальної мікроенергостанції зі схожою архітектурою, у якій програмні агенти будуть працювати на відповідних мікроконтролерах. Це, а також організація управління режимами роботи мікроенергостанції із застосуванням енергоринку та модифікованого DAM-алгоритму є перевагою розробленого програмного комплексу у порівнянні із аналогами.

Методи використані у програмній реалізації спрощують роботу по обробці даних, полегшують масштабованість коду та забезпечують належну, не надлишкову зв'язність компонентів коду.

Розроблена система може використовуватись у будь-якому середовищі, де важливим є питання енергозбереження та оптимального використання енергоресурсів, але першим і основним її користувачем має стати кафедра. Система та її компоненти стануть базою для майбутніх розробок, удосконалень та науково-дослідних робіт.

Як комерційний продукт система може бути використана у галузі сучасної енергетики, в якості інструмента асистування при побудові мікроенергостанцій.

4. СТАРТАП ПРОЕКТ

Стартап — це нещодавно створена компанія, яка знаходиться лише на стадії розвитку, але має інноваційні ідеї. Саме ці нововведення допоможуть перевершити конкурентів та вийти на ринок з новим продуктом, послугою. Основна особливість стартапів – недостатність фінансів, пошук інвесторів та відношення не тільки до ІТ-сфери, а і до будь-якої області ринку.

Стартап як форма малого ризикового (венчурного) підприємництва впродовж останнього десятиліття набула широкого розповсюдження у світі через зниження бар'єрів входу в ринок (із появою Інтернету як інструменту комунікацій та збуту стало простіше знаходити споживачів та інвесторів, займатись пошуком ресурсів, перетинати кордони між ринками різних країн), і вважається однією із наріжних складових інноваційної економіки, оскільки за рахунок мобільності, гнучкості та великої кількості стартап-проектів загальна маса інноваційних ідей зростає. Створення та ринкове впровадження стартап-проектів відзначається підвищеною мірою ризику, ринково успішними стає лише невелика частка, що за різними оцінками складає від 10% до 20%. Ідея стартап-проекту, взята окремо, не вартує майже нічого: головним завданням керівника проекту на початковому етапі його існування є перетворення ідеї проекту у працюючу бізнес-модель, що починається із формування концепції товару (послуги) для визначеної клієнтської групи за наявних ринкових умов.

У даному розділі описано стартап-проект, що відповідає виконаній задачі дипломної роботи.

4.1 Опис ідеї проекту

У цьому та наступних пунктах розділу проаналізовані дані, відповідно до плану розробки стартап-проекту подано у вигляді таблиць. У таблиці 4.1 наведено

зміст ідеї проекту, напрямки застосування та основні вигоди користувача. Ці характеристики дають цілісне уявлення про сутність проекту та потенційні ринки, в межах яких слід шукати клієнтів.

Таблиця 4.1 — Опис ідеї стартап проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувачів
Надати можливість моделювання та симуляції мікроенергостанцій як інструменту асистування при їх проектуванні сучасним енергетичним компаніям	Моделювання та симуляція роботи мікроенергостанцій у відповідності з децентралізованими системами енергоменеджменту	Можливість проектування та перевірки економічних показників використання децентралізованої мультиагентної системи енергоменеджменту мікроенергостанції
	Використання продукту та програмного агента у якості системи енергоменеджменту існуючої мікроенергостанції	Покращення економічних показників існуючої системи енергоменеджменту за рахунок автоматизації

Продукт відрізняється від існуючих аналогів та замінників мультиагентним підходом, гібридною архітектурою системи енергоменеджменту, застосуванням енергоринку та модифікованого DAM-алгоритму його роботи.

Наступний крок передбачає аналіз потенційних техніко-економічних переваг ідеї, порівняно із пропозиціями конкурентів (таблиця 4.2).

Визначений перелік слабких, сильних та нейтральних характеристик та властивостей ідеї потенційного товару є підґрунтям для формування його конкурентоспроможності.

Таблиця 4.2 — Визначення сильних, слабких та нейтральних характеристик

№ п/п	Техніко- економічні характеристики ідеї	Товари/концепції конкурентів		W	N	S
		Розроблений продукт	DER-CAM			
1	Економічні	Прибуток з користувачів				
2	Призначення	Моделювання та симуляція із застосуванням мультіагентного підходу	Моделювання та розрахунок економічних характеристик			+
3	Надійності	Безвідмовна робота				
4	Ергономічні	Зручний користувацький інтерфейс, інтуїтивне налаштування конфігурації	Неінтуїтивне розташування елементів інтерфейсу			+
5	Безпеки	Захист інформації користувачів				

У таблиці 4.2, значення W, N, S відповідають слабким, нейтральним та сильним характеристикам системи у порівнянні з найближчим конкурентом на ринку — DER-CAM.

4.2 Технологічний аудит ідеї проекту

Було проведено аудит технології, за допомогою якої можна реалізувати

ідею проекту (технології створення товару). Аналіз складових технологічної здійсненності ідеї проекту наведено у таблиці 4.3.

Таблиця 4.3 — Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології реалізації	Наявність технологій	Доступність технологій
1	Програмне моделювання мікроенергостанції	Інструменти побудови мультиагентних моделей	+	+
2	Децентралізована система енергоменеджменту із гібридною архітектурою	Алгоритми утворення та функціонування енергоринку	+	+
3	Програмний агент моніторингу та управління режимами роботи мікроенергостанції	Інструменти побудови мультиагентних моделей, розв’язувані задач оптимізації	+	+
4	Навчання програмних агентів	Алгоритми та інструменти машинного навчання	+	+

За результатами аналізу зроблено висновок щодо можливості технологічної реалізації проекту. Технологічним шляхом реалізації проекту було обрано такі технології, як Repast для мультиагентного моделювання, QuadProg для вирішення задачі оптимізації та JReLM для навчання агентів.

4.3 Аналіз ринкових можливостей запуску стартап проекту

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів.

Спочатку було проведено аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (таблиця 4.4).

Таблиця 4.4 — Попередня характеристика потенційного ринку стартап проекту

№ п/п	Показники стану ринку	Характеристика
1	Кількість головних гравців, од	Більше 20
2	Загальний обсяг продаж	\$10 млрд
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу	Відсутні
5	Специфічні вимоги до стандартизації та сертифікації	Відсутні
6	Середня норма рентабельності в галузі (або по ринку), %	38

Середню норму рентабельності в галузі було порівняно із банківським відсотком на вкладення. Останній є меншим, тому є сенс вкладати гроші саме у цей проект.

За результатами аналізу таблиці 4.4 було зроблено висновок, що ринок є привабливим для входження.

Далі були визначені потенційні групи клієнтів, їх характеристики та сформовано орієнтовний перелік вимог до товару для кожної групи (таблиця 4.5).

Таблиця 4.5 — Характеристика потенційних клієнтів стартап проекту

№ п/п	Потреба що формує ринок	Цільова аудиторія	Відмінності у поведінці різних цільових груп клієнтів	Вимоги споживачів до товару
1	Потреба у програмному забезпеченні для моделювання мікроенергостанцій	Архітектори систем у галузі енергетики, студенти та викладачі відповідних спеціальностей	Використання для моделювання та вивчення особливостей DEMS	Зручний інтерфейс, точність результатів, швидкість, надійність у використанні
2	Організація автоматизованого управління мікроенергостанціями	Власники та розробники мікроенергостанцій	Потребують надійний інструмент для уникнення економічних збитків	

Після визначення аудиторії та потенційних груп клієнтів було проведено аналіз ринкового середовища: складено таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (таблиці 4.6, 4.7). Основними такими факторами зазвичай є фінансування, конкуренція та конкурентоспроможність продукту, проблеми надійності, швидкості розробки, вартості експлуатації тощо.

Таблиця 4.6 — Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Фінансування	Проблема де взяти гроші, особливо на початковій стадії запуску проекту	Пошук інвесторів
2	Конкуренція	Низький інтерес до сервісу та недостатньо велика кількість користувачів	Передбачити додаткові переваги власного програмного продукту
3	Реклама	Можливі складнощі в розробці маркетингової кампанії та укладенні відповідних угод	Реклама для збільшення аудиторії користувачів
4	Невідповідність умовам технологічного розвитку	Динамічний розвиток технологій, що призведе до невідповідності ПП використовуваним у сучасних приміщеннях технологіям	Забезпечення гнучкості математичних моделей, адаптація до сучасних умов швидкими темпами
5	Розробка функціоналу	Можливі складнощі при розробці, відставання від термінів тощо	Найняти додатковий персонал, змінити терміни тощо
6	Запуск та супровід	Можливі непередбачені проблеми	Створити команду технічної підтримки
7	Зміна потреб користувачів	Користувачам необхідне програмне забезпечення з іншим функціоналом	Передбачити можливість додавання нового функціоналу до створюваного ПП

Таблиця 4.7 — Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Прибуток	Отримання прибутку	Заходи для збільшення рентабельності
2	Конкуренція	Відсутність аналогічного продукту для вітчизняного користувача	Адаптація програмного продукту до вітчизняних особливостей
3	Поява нових методів моделювання	З'являються нові методи побудови програмних мультиагентних моделей	Покращити ПП додаванням нового функціоналу, розширення можливостей
4	Поява нових методів оптимізації	З'являються нові методи розрахунку, що дозволитимуть враховувати більше факторів	Покращити ПП додаванням нового функціоналу, розширення можливостей

Наступним етапом було проведення аналізу пропозиції: визначено загальні риси конкуренції на ринку (таблиця 4.8).

Таблиця 4.8 — Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється характеристика	Вплив на діяльність підприємства
1. Конкуренція: чиста	На ринку присутні декілька компаній-конкурентів, але їх товар дещо відрізняється між собою	Підтримка якості продукту та постійні нововведення, вдосконалення.

Особливості конкурентного середовища	В чому проявляється характеристика	Вплив на діяльність підприємства
2. Рівень конкурентної боротьби: національний	Компанії-конкуренти національного рівня	Створити ПП з урахуванням потреб користувачів у нашій країні
3. За галузевою ознакою: внутрішньогалузева	Продукт може використовуватись у енергетичній галузі	Постійне вдосконалення продукту з метою його застосування у інших сферах
4. Конкуренція за видами товарів: товарно-видова	Конкуренція між видами ПП, їх особливостями	Створити ПП, враховуючи недоліки конкурентів
5. Характер конкурентних переваг: неціновий	Вдосконалення технології створення ПП, для зниження ціни і розширення функціональності	Використання більш дешевих технологій для розробки, ніж використовують конкуренти, але тільки за їх відповідності вимогам якості
6. Інтенсивність: не марочна	Бренд присутній, але його роль незначна	Реклама, участь у конференціях, семінарах

Було проведено аналіз конкуренції у галузі за моделлю М. Портера (таблиця 4.9).

За результатами аналізу таблиці 4.9 було зроблено висновок про можливість роботи на ринку з огляду на конкурентну ситуацію. Також було зроблено висновок щодо характеристик, які повинен мати проект, щоб бути конкурентоспроможним на ринку.

Таблиця 4.9 — Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари замітники
	DER-CAM, HOMER	Бар'єри входження в ринок: гнучкість цін, розмір капіталовкладень, зв'язки з рекламодавцями, широкий спектр можливостей	-	Фактори сили: змінні витрати, високий рівень чутливості до зміни цін, контроль якості	Фактори загроз: ціна, лояльність споживачів
Висновки	Висока інтенсивність конкурентної боротьби	є можливість входу в ринок, є потенційні конкуренти, строки виходу їх на ринок: 0.5-1 рік	-	Для клієнтів найбільш важлива цінова доступність послуг та якість послуг	Обмеження для роботи на ринку через товари замітники: обмеження економічної рентабельності продукту

На основі аналізу конкуренції, наведеного в таблиці 4.9, а також з урахуванням характеристик ідеї проекту (таблиця 4.2), вимог споживачів до товару (таблиця 4.5) та факторів маркетингового середовища (таблиці 4.6, 4.7) визначається та обґрунтовується перелік факторів конкурентоспроможності. Аналіз наведено у таблиці 4.10.

Таблиця 4.10 — Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор	Обґрунтування (перелік чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Ціна	Доступніша ціна збільшує кількість потенційних клієнтів
2	Орієнтованість на кінцевого користувача	Продукт орієнтований на взаємодію з клієнтом
3	Розширені можливості у порівнянні із конкурентами	Продукт дозволяє не лише моделювати мікроенергостанції, а і виконувати симуляцію їх роботи, а також застосовувати компоненти змодельованої системи у реальній мікроенергомережі

За визначеними факторами конкурентоспроможності (таблиця 4.10) проведено аналіз сильних та слабких сторін стартап-проекту (таблиця 4.11).

Таблиця 4.11 — Порівняльний аналіз сильних та слабких сторін проекту

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з розроблюваним проектом						
			-3	-2	-1	0	+1	+2	+3
1	Ціна	13			+				

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з розроблюваним проектом						
2	Орієнтованість на кінцевого користувача	10				+			
3	Розширені можливості у порівнянні із конкурентами	20		+					

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities) (таблиця 4.12) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (таблиця 4.11).). Перелік ринкових загроз та ринкових можливостей було складено на основі аналізу факторів загроз та факторів можливостей маркетингового середовища.

Таблиця 4.12 — Ступеневий аналіз конкуренції на ринку

Сильні сторони: ціна орієнтованість на кінцевого користувача ширший функціонал за продукти аналоги	Слабкі сторони: можливе відставання від розвитку предметної області складність алгоритмів
Можливості: конкуренція поява нових методів моделювання та оптимізації	Загрози: невідповідність умовам технічного розвитку зміна потреб користувачів

Ринкові загрози та ринкові можливості є наслідками (прогнозованими

результатами) впливу факторів, і, на відміну від них, ще не є реалізованими на ринку та мають певну ймовірність здійснення. Зниження доходів потенційних споживачів є фактором загрози, на основі якого можна зробити прогноз щодо посилення цінового фактору при виборі товару та цінової конкуренції.

На основі SWOT-аналізу було розроблено альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок. Визначені альтернативи були проаналізовані з точки зору ймовірності отримання ресурсів (таблиця 4.13).

Таблиця 4.13 — Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Терміни реалізації
1	Безкоштовне розповсюдження створеного ПП	35%	12 міс.
2	Створення ПП з подальшим розповсюдженням за певну оплату	80%	18 міс.
3	Створення вебсайту, на якому можна буде користуватися ПП	70%	16 міс.

Для подальшої реалізації було обрано альтернативу №2, за якої отримання ресурсів є більш простим та ймовірним.

4.4 Розроблення ринкової стратегії стартап проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: було проведено опис цільових груп потенційних споживачів (таблиця 4.14).

Таблиця 4.14 — Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтов- ний попит в межах цільової групи (сег- менту)	Інтенсив- ність кон- куренції в сегменті	Простота входу у сегмент
1	Компанії енергетичної галузі	Висока	Високий	Висока	Помірна
2	Студенти та викладачі енергетичних спеціальностей	Висока	Помірний	Помірна	Висока
3	Архітектори та розробники мікроенерго- станцій та систем енерго- менеджменту	Висока	Помірний	Висока	Низька

За результатами аналізу потенційних груп споживачів було обрано цільові групи, для яких буде запропоновано товар та визначено стратегію охоплення ринку — стратегію диференційованого маркетингу (компанія працює з декількома сегментами).

Для роботи в обраних сегментах ринку сформовано базову стратегію розвитку (таблиця 4.15).

Таблиця 4.15 — Фактори можливостей

Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Створення ПП з подальшим розповсюдженням за певну оплату	Визначити потреби кожної з груп, розробити відповідно до них стратегії приваблення клієнтів та маркетингової комунікації	Цінова політика, універсальність продукту (миттєве практичне застосування), орієнтованість на кінцевого користувача	Стратегія диференціації

Наступним кроком обрано стратегію конкурентної поведінки (таблиця 4.16).

Таблиця 4.16 — Визначення базової стратегії конкурентної поведінки

Чи є проект першопрохідцем на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
Частково	Так	Так. Такі як ціна послуг	Стратегія виклику лідера

На основі вимог споживачів з обраних сегментів до постачальника (стартап

компанії) та до продукту (див. таблицю 4.5), а також в залежності від обраної базової стратегії розвитку (таблиця 4.15) та стратегії конкурентної поведінки (таблиця 4.16) розроблено стратегію позиціонування (таблиця 4.17), що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати торгівельну марку/проект.

Таблиця 4.17 — Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власно-го проекту.
1	Цінова доступність	Наступальна стратегія	Нижча ціна за рахунок використання доступних технологій	Економія часу, зручність застосування, практичність та точність результату
2	Легкість розуміння, зручний інтерфейс, надійний, швидкий та точний продукт	Стратегія диференціації	Застосування мультиагентного підходу для моделювання та організації роботи мікроенергостанції	

В результаті отримано узгоджену систему рішень щодо ринкової поведінки стартап-компанії.

4.5 Розроблення маркетингової програми стартап проекту

Сформовано маркетингову концепцію товару, який отримає споживач. Для цього у таблиці 4.18 підсумовано результати попереднього аналізу конкурентоспроможності товару.

Таблиця 4.18 — Визначення ключових переваг концепції товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
1	Моделювання та симуляція мікроенергостанцій	Можливості мультиагентного моделювання	Можливість симуляції змодельованих систем, полегшена
2	Автоматизація режимів роботи мікроенергостанції	Застосування децентралізованої системи енергоменеджменту гібридної архітектури та програмного агента моніторингу та управління режимами роботи мікроенергостанції	інтегрованість в існуючі мікроенергостанції, впровадження автоматизації і, відповідно, підвищення економічної ефективності області застосування програмного продукту
3	Інтегрованість в існуючі мікроенергостанції	Можливість застосування програмних агентів на мікроконтролерах в існуючих мікроенергостанціях	

Розроблено трирівневу маркетингову модель товару: уточнення ідеї продукту та/або послуги, його фізичні складові, особливості процесу його надання (таблиця 4.19).

Таблиця 4.19 — Опис трьох рівнів моделі товару

Рівень товару	Сутність та складові
Товар за задумом	Система моделювання мікроенергостанцій та програмний агент моніторингу та управління режимами роботи мікроенергостанцій
Товар у реальному виконанні	Можливості моделювання, симуляції змінних конфігурацій мікроенергостанцій
Товар із підкріпленням	Зручний графічний інтерфейс користувача, можливості візуалізації даних
За рахунок чого товар буде захищено від копіювання: програмним шляхом	

Після формування маркетингової моделі товару слід відмітити, що проект буде захищено від копіювання шляхом патентування. Наступним кроком є визначення цінових меж (таблиця 4.20), якими необхідно керуватись при встановленні ціни на потенційний товар.

Таблиця 4.20 — Визначення меж встановлення ціни

Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
2000-14000 грн	близько 20000 грн.	всі	2000-20000 грн

Остаточне визначення ціни відбувається під час фінансово-економічного аналізу проекту, яке передбачає аналіз ціни на товари-аналоги або товари субститути, а також аналіз рівня доходів цільової групи споживачів.

Аналіз проведено експертним методом. В результаті проведення аналізу встановлено верхню та нижню межі встановлення ціни на товар, що надається користувачам.

Наступним кроком є визначення оптимальної системи збуту (таблиця 4.21).

Таблиця 4.21 — Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Корпоративне та індивідуальне споживання товарів та послуг	Збут здійснюється власними силами через магазини цифрової дистрибуції	Однорівневий канал	Продукт — користувач

Останньою складовою маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (таблиця 4.22).

Результатом аналізу стала ринкова (маркетингова) програма, що включає в себе концепції товару, збуту, просування та попередній аналіз можливостей ціноутворення, спирається на цінності та потреби потенційних клієнтів, конкурентні переваги ідеї, стан та динаміку ринкового середовища, в межах якого впроваджено проект, та відповідну обрану альтернативу ринкової поведінки.

Таблиця 4.14 — Вибір цільових груп потенційних споживачів

Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
Користувачі потребують можливості використовувати функціонал продукту	Конференції, інтернет-конференції, семінари, огляд професійної літератури, інтернет, періодичні видання у різноманітних (профільних) галузях	Позиція на основі порівняння фірми з товарами конкурентів; Особливості потреб споживачів	Зацікавити користувачів	Переконати користувачів, що функціонал їм потрібен для забезпечення продуктивної роботи мікроенергостанцій

Висновки до розділу 4

Проведено аналіз програмного продукту з позиції перспективи розвитку стартап-проекту, стан ринку існуючих рішень, а також потреб потенційних користувачів. Запропоновано межі цінового формування для розробленої програмної системи. Визначено стратегії просування стартап-проекту. Розроблено маркетингову програму.

ВИСНОВКИ

— Визначено недоліки існуючих аналогів: обмежені або відсутні засоби симуляції моделі, а також відсутність моделювання взаємодії між компонентами мікроенергостанції. В результаті аналізу підходів до побудови інформаційних систем управління мікроенергостанціями обрано гібридну децентралізовану модель архітектури та енергоринок як засіб організації управління. Використання децентралізованої архітектури дозволяє відмовитись від дорогого обчислювального та комунікаційного обладнання, спрощує підтримку та модифікацію системи. Енергоринок у якості підходу до управління мікроенергостанцією, забезпечує гнучкість системи, керованість попитом та її адаптованість до змін у зовнішньому середовищі.

— Розроблені програмні моделі функціонування елементів інфраструктури мікроенергостанцій за рахунок уніфікованості спрощують розробку та масштабування мультиагентної системи, враховуючи при цьому індивідуальні особливості кожного елемента через його енергетичні характеристики. Агенти-генератори було вдосконалено впровадженням стратегічної поведінки на ринку із застосуванням машинного навчання з підкріпленням. Це призвело до більш ефективної експлуатації генераторів, а також збільшення їх прибутків на 40%.

— Розроблений продукт дозволяє реалізацію сценаріїв управління з урахуванням мінливості середовища, що дозволяє підвищити технологічну та економічну ефективність мікроенергостанцій.

Програмний продукт має переваги над існуючими аналогами та є конкурентноздатним на ринку. Визначено можливі напрямки вдосконалення системи: реалізація нових алгоритмів функціонування енергоринку, розширення можливостей конфігурації агентів, введення в систему нових агентів.

Основна цільова аудиторія — компанії енергетичної галузі, а також кафедри енергетичних спеціальностей університетів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Microgrids / N. Hatziargyriou, H. Asano, R. Iravani, C. Marnay. // IEEE Power Energy Mag. — 2007. — №5. — С. 78–94.
2. Microgrids management / F.Katiraei, R. Iravani, N. Hatziargyriou, A. Dimeas. // IEEE Power Energy Mag. — 2008. — №6. — С. 78–94.
3. Karavas C. Renewable energy driven small-scale sea water reverse osmosis desalination systems: A survey. / C. Karavas, G. Papadakis. // Fundam. Renew. Energy Appl. — 2017. — №7. — С. 115.
4. Optimal dispatching of distributed generators and storage / S.Conti, R. Nicolosi, S. Rizzo, H. Zeineldin. // IEEE Trans. Power Deliv. — 2012. — №27. — С. 1243–1251.
5. Lidula N. Microgrids research: A review of experimental microgrids and test systems / N. Lidula, A. Rajapakse. // Renew. Sustain. Energy Rev. — 2011. — №15. — С. 186–202.
6. Design and implementation of CCNY DC microgrid testbed / M.Saleh, Y. Esa, Y. Mhandi, W. Brandauer. // IEEE Industry Applications Society Annual Meeting. — 2016. — №7.
7. Dynamic operation and control of a hybrid nanogrid system for future community houses / P.Chandrasena, S. Ruwan, Shania, Farhad. // IET Generation, Transmission & Distribution. — 2015. — №9. — С. 1168–1178.
8. Ali L. Determination of an economically-suitable and sustainable standalone power system for an off-grid town in Western Australia / L. Ali, F. Shania. // Renewable Energy. — 2017. — №106. — С. 243–254.
9. Pashajavid E. Development of a Self-Healing Strategy to Enhance the Overloading Resilience of Islanded Microgrids / E. Pashajavid, F. Shahnian, A. Ghosh. // IEEE Transactions on Smart Grid. — 2015. — №1.
10. Emily P. U.S. Strategic Solar Photovoltaic-Powered Microgrid Deployment for

- Enhanced National Security / P. Emily, S. Chelsea, P. Joshua. // Renewable Energy. — 2017. — №92.
11. The VERITAS multi-technology microgrid experiment: a design review / M. Guarnieri, A. Bovo, A. Giovannelli, P. Mattavelli. // IEEE Industrial Electronics Magazine. — 2018. — №12. — C. 19–31.
 12. Jin M. MOD-DR: Microgrid optimal dispatch with demand response / M. Jin, W. Feng, P. Liu. // Applied Energy. — 2017. — №187. — C. 758–776.
 13. Prerna G. Investigations on Issues in Microgrids / G. Prerna, S. Sunita. // 5. — 2017.
 14. Xiao Z. An overview of microgrid control / Z. Xiao, J. Wu, N. Jenkins. // Intelligent Automation and Soft Computing. — 2010. — №16. — C. 199–212.
 15. Tan X. Advances and trends of energy storage technology in microgrid / X. Tan, Q. Li, H. Wang. // Int. Journal of Electrical Power & Energy Syst. — 2013. — №44. — C. 179–191.
 16. Feng W. Mitigation of harmonics of DFIGs in DC microgrids / W. Feng, M. Vilathgamuwa, S. Choi. // in Proc. IEEE Energy Convers. — 2012. — №15
 17. Moore T. Challenges and opportunities for inverters in microgrids / T. Moore, S. Sayeef. // IEEE Industrial Electronics Society. — 2011. — №7.
 18. Microgrid supervisory controllers and energy management systems: A literature review / L. Meng, E. Sanseverino, A. Luna, T. Dragicevic. // Renew. Sustain. Energy Rev. — 2016. — №60. — C. 1263–1273.
 19. Ilić M. D. Hierarchical power systems control: Its value in a changing industry / M. D. Ilić, S. X. Liu., 1996. — 241 c. — (University of Michigan).
 20. Marnay C. Microgrids in the evolving electricity generation and delivery infrastructure/ Chris Marnay, Giri Venkataramanan// 2006 IEEE Power Engineering Society General Meeting. — 2006. — 5p.
 21. Ilic-Spong M. Secondary voltage control using pilot point information/ Marija Ilic-Spong, Juha Christensen, K. L. Eichorn// Power Syst., IEEE Trans. — 1988. — Vol. 3. — pp. 660-668.

22. Rahimian M.A Review of Predictive Software for the Design of Community Microgrids/ Mina Rahimian, Lisa D. Iulo, Jose M. Pinto Duarte// Journal of Engineering. — 2018. — Vol. 2018. — 13 p. [Електронний ресурс]. – Режим доступу: <https://www.hindawi.com/journals/je/2018/5350981/>
23. DER-CAM User Manual/ Berkeley, California: Lawrence Berkeley National Laboratory (LBNL) — 2015. — 55p.
24. Lambert T. Micropower system modeling with HOMER/Tom Lambert, Paul Gilman, Peter Lilienthal//Integration of Alternative Sources of Energy. — 2006. — Vol. 1, № 15,— pp. 379–418.
25. Використання HMI/SCADA-системи zenon в енергетиці [Електронний ресурс]. – 2017. – Режим доступу до ресурсу: http://www.svaltera.ua/catalogs/knowledge-base/brands/copa_data/zenon_in_energy
26. Ferber J. Multi-agent systems: an introduction to distributed artificial intelligence/ Jacques Ferber — Boston, Addison-Wesley Longman Publishing, 1999. — 528p.
27. Macal C.M. Tutorial on agent-based modelling and simulation/ Charles M. Macal, Michael John North// Journal of Simulation. — 2010. — Vol. 4(3). — pp. 151–162.
28. Wooldridge M.J. An introduction to multiagent systems/ Michael Wooldridge. — Wiley, 2009. — 484p.
29. Ferber J. Les systèmes multi-agents: vers une intelligence collective/ Jacques Ferber. — InterEditions, 1995. — 522p.
30. Ehrentreich N. Agent-Based Modeling : The Santa Fe Institute Artificial Stock Market Model Revisited(Lecture Notes in Economics and Mathematical Systems)/ Norman Ehrentreich — Springer-Verlag Berlin Heidelberg, 2008. — 231p.
31. Poslad S. Specifying Protocols for Multi-Agent Systems Interaction / Stefan Poslad. // ACM Transactions on Autonomous and Adaptive Systems. — 2007. — №4.
32. Bellifemine F. Developing multi-agent systems with JADE / F. Bellifemine, G. Caire, D. Greenwood. – Chichester: Wiley, 2007. — 303 с.

33. Collier N. RePast: An Extensible Framework for Agent Simulation [Электронный ресурс] / Nick Collier // University of Chicago. — 202. — Режим доступа до ресурсу: <http://www2.econ.iastate.edu/tesfatsi/RepastTutorial.Collier.pdf>.
34. Energy Management Algorithm for Smart Home with Renewable Energy Sources / A.Boynuegri, M. Baysal, B. Yagciteking, A. Karakas. // Power Engineering, Energy and Electrical Drives. — 2013. — №10.
35. Realistic electricity market simulator for energy and economic studies / J.Bernal-Agustin, J. Contreras, R. Martin-Flores, A. Conejo. // Electric Power Systems Research. — 2007. — №77. — С. 46–54.
36. DCOPFJ (Java): A Free Open-Source Solver for Bid/Offer-Based DC Optimal Power Flow Problems [Электронный ресурс] // Iowa State University. — 2019. — Режим доступа до ресурсу: <http://www2.econ.iastate.edu/tesfatsi/DCOPFJHome.htm>.
37. Goldfarb D. A Numerically Stable Dual Method for Solving Strictly Convex Quadratic Programs/ Donald Goldfarb, Ashok Udhawdas Idnani// Mathematical Programming. — 1983. — Vol. 27. — pp. 1-33.
38. Erev I. Predicting How People Play Games: Reinforcement Learning in Experimental Games with Unique, Mixed Strategy Equilibria/Ido Erev, Alvin E. Roth//The American Economic Review. — 1998. Vol. 88, №4. — pp. 848–881.
39. Nicolaisen J. Market power and efficiency in a computational electricity market with discriminatory double-auction pricing/James Nicolaisen, Valentin Petrov, Leigh Tesfatsion//IEEE Transactions on Evolutionary Computing. — 2001. — Vol. 5, № 5. — pp. 504–523.

ДОДАТОК А

Розробка програмного агента моніторингу та управління режимами роботи
мікроенергостанцій

Апробація

УКР.НТУУ”КПІ”_ТЕФ_АПЕПС_ТВ42133_19М

Аркушів 3

Київ — 2019

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

СУЧАСНІ ПРОБЛЕМИ НАУКОВОГО ЗАБЕЗПЕЧЕННЯ ЕНЕРГЕТИКИ

Матеріали XVII Міжнародної
науково-практичної конференції
молодих вчених та студентів
м. Київ, 23-26 квітня 2019 року,

ТОМ 2



Київ- 2019

Використання методів виявлення автоматичних моделей поведінки для побудови аналітичних сценаріїв.

ОЛЕНЄВА К.М., аспірант;

ШПУРИК В.В.

Керівник - доц., к.т.н. Коваль О.В.

Візуалізації структури KNX-мережі з використанням людинно-машинного інтерфейсу.

ЯШИН А.С., магістрант гр. ТМ-81мп

Керівник - доц., к.е.н. Сегеда І.В.

Розробка агента моніторингу і управління попитом на електричну енергію "Розумної будівлі".

ШАРНІН С.А., магістрант гр. ТІ-81мп

Керівник - доц., к.т.н. Ковальчук А.М.

Агент моніторингу та управління режимами роботи мікроенергостанцій.

СТОЛЯР А.В., магістрант гр. ТВ-81мп

Керівник - доц., к.т.н. Ковальчук А.М.

Розробка серверної частини для веб-додатку відкритий спортивний майданчик з е-сервісами.

СЕРБІН А.В., магістрант гр. ТВ-82

Керівник - доц., к.т.н. Ковальчук А.М.

Розв'язок задачі обрахунку водонагрівача в інтерактивному режимі з використанням клієнт-серверної архітектури.

РОМАНОВ О.В., магістрант гр. ТВ-61мп

Керівник - доц., к.т.н. Кузьменко І.М.

Оцінка територій для побудови вітроелектростанцій із застосуванням мультиагентних технологій та ГІС.

ПІДВИШЕННИЙ Т.О., магістрант гр. ТВ-81мп

Керівник - асист. Швайко В.Г.

REST-інтерфейс як основа комунікації систем контролю доступу.

ОБРУСНІК Д.В., студент гр. ТВ-82мп

Керівник - доц., к.е.н. Левченко Л.О.

Інструментальні засоби аналізу впливу параметрів експерименту на сигнатуру морського об'єкту.

ОБІЩЕНКО А.А., магістрант гр. ТВ-81мп

Керівник - доц., к.т.н. Варава І.А.

Система моделювання структури та функціонального контенту інженерних систем енергоефективної будівлі.

КУРСЕНКО Л.О., магістрант гр. ТІ-81мп

Керівник - доц., к.т.н. Шпурик В.В.

Інструментальний засіб підтримки динамічного реєстру інформаційних ресурсів на базі ОРБД Caché.

КОСТЕНКО О.П., магістрант гр. ТВ-81мп

Керівник - доц., к.т.н. Кублій Л.І.

Мультиагентні системи в децентралізованих мережах енергоспоживання.

ЖОРНОВИЙ Е.Г., магістрант гр. ТВ-82

Керівник - доц., к.т.н. Ковальчук А.М.

Розробка веб-додатку для відкритого спортивного майданчику з е-сервісами.

АМБРОС С.М., магістрант гр. ТВ-81мп

Керівник - доц., к.т.н. Ковальчук А.М. .

УДК 004.78

Магістрант 5 курсу, гр. ТВ-81мп Столяр А.В.
Доц., к.т.н. Ковальчук А.М.

АГЕНТ МОНІТОРИНГУ ТА УПРАВЛІННЯ РЕЖИМАМИ РОБОТИ МІКРОЕНЕРГОСТАНЦІЙ

Зростання чисельності населення, урбанізація та модернізація призвели до однієї з найбільших проблем сучасності, яка полягає у значному зростанні світового попиту на електроенергію. Економічно вигідним вирішенням проблеми глобального споживання енергетичних ресурсів є застосування мікроенергостанцій або енергетичних мікромереж.

Мікроенергостанції являють собою мережі низької або середньої напруги, які можуть інтегрувати відновлювані джерела енергії (ВДЕ), генератори, системи зберігання енергії та енергоспоживання, тобто являють собою системи енергетичної інфраструктури.

Мікроенергостанції можуть працювати як в автономному режимі, так і в режимі підключення до глобальної енергомережі. У режимі, підключення до глобальної мережі, мікроенергостанція закуповує електроенергію з мережі або вводить в неї енергію, щоб регулювати баланс потужності між споживанням енергії та навантаженням і максимізувати її експлуатаційні переваги. В автономному режимі система виробляє та надає електроенергію споживачам, зберігаючи баланс між стабільністю виробництва та споживанням.

Для забезпечення ефективної роботи мікроенергостанцій необхідні системи управління режимами їх роботи, або системи управління енергією (EMS), що також важливо для застосування оптимальних стратегій управління. EMS енергетичних мікромереж можуть бути розділені на два основні класи: централізовані EMS (CEMS) і децентралізовані EMS (DEMS). DEMS архітектура є більш гнучкою і менш складною, оскільки вона заснована на мережі автономних локальних контролерів. Компоненти децентралізованої системи є незалежними, формуючи інтелектуальну і динамічну систему, що забезпечує енергоефективність і відповідає змінним умовам, таким як мінливість ВДЕ або навантаження. У DEMS можуть використовуватись передові методи управління завдяки розвитку технології мікроконтролерів.

Агент моніторингу і управління режимами роботи мікроенергостанцій є одним із найважливіших компонентів децентралізованої системи управління енергією. Являючи собою абонента для зовнішньої енергетичної мережі, до якої інтегрується енергетична мікромережа, він об'єднує усіх її агентів під єдиною "парасолькою". Агент здатен відстежувати зміни як в зовнішній енергетичній мережі, так і в мікромережі, та впливати на неї для досягнення цілей свого проектування. Паралельно взаємодіючи, координуючи або ведучи переговори з іншими агентами системи, він обробляє отриману інформацію, порівнює з певними правилами, досліджує можливі рішення і, нарешті, вибирає оптимальне рішення, таким чином реалізуючи концепцію Demand Side Management.

Метою розробки агента моніторингу та управління режимами роботи мікроенергостанцій є створення програмного та апаратного модулів у складі системи енергетичної інфраструктури із врахуванням недоліків існуючих аналогічних програмних рішень. Розроблювана реалізація має відповідати ключовим вимогам до інтелектуального агента, бути архітектурно гнучкою, відносно недорогою та мати можливості до адаптації та застосування у різних мікроенергомережах.

Перелік посилань:

1. Hatziaargyriou, N.; Asano, H.; Iravani, R.; Marnay, C. Microgrids. IEEE Power Energy Mag. 2007, Vol. 5.
2. Hatziaargyriou, N.; Dimeas, A.; Tsikalakis, A. Centralized and decentralized control of microgrids. Int. J. Distrib. Energy Resour. 2005, Vol. 1.